1995-03

# Proposed management architecture for managing a distributed network information system

Everingham, Michael A.

Monterey, California. Naval Postgraduate School

# NAVAL POSTGRADUATE SCHOOL
## Monterey, California




# THESIS

**PROPOSED MANAGEMENT ARCHITECTURE FOR MANAGING A DISTRIBUTED NETWORK INFORMATION SYSTEM**

by

Michael A. Everingham

March 1995

Principal Advisor: Myung Suh

**Approved for public release; distribution is unlimited.**

# REPORT DOCUMENTATION PAGE

*From Approved OMB No. 0704-0188*



| 1. AGENCY USE ONLY *(Leave blank)* | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | March 1995 | Master's Thesis |

**4. TITLE AND SUBTITLE**
PROPOSED MANAGEMENT ARCHITECTURE FOR MANAGING A DISTRIBUTED NETWORK INFORMATION SYSTEM

**5. FUNDING NUMBERS**

**6. AUTHOR(S)**
Everingham, Michael A.

**7. PERFORMING ORGANZATION NAME(S) AND ADDRESS(ES)**
Naval Postgraduate School
Monterey, CA 93940-5000

**8. PERFORMING ORGANIZATION REPORT NUMBER**

**9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

**11. SUPPLEMENTARY NOTES**
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

**12a. DISTRIBUTION / AVAILABILITY STATEMENT**
Approved for public release; distribution is unlimited

**12b. DISTRIBUTION CODE**



**13. ABSTRACT** *(Maximum 200 words)*

Today's organizations are becoming increasingly dependent on their information infrastructure. Whether it is an effort to maintain their competitive advantage, or provide a Joint Task Force Commander with a real-time picture of the battlefield, the organization's information infrastructure is becoming a vital key to the organization's success or failure in performing its mission-critical processes. "However, the complexity of the modern infrastructure, and the changes affecting it, produce significant management challenges." [Ref. 1, p. 451]

The purpose of this study is to develop a management architecture that will assist Network Managers in formulating network management strategies that will be flexible, proactive, distributed, standardized, comprehensive, collaborative and integratable. It also lays the foundation for the development of an Management Information Base (MIB), specifically that portion which relates to Asset Management.

The need for a comprehensive network management architecture is clear. The proposed

**14. SUBJECT TERMS**
Management, Network Management, Asset Management, Database Design, MIB Design, Information Systems, Management Framework, Management Architecture

**15. NUMBER OF PAGES**
82

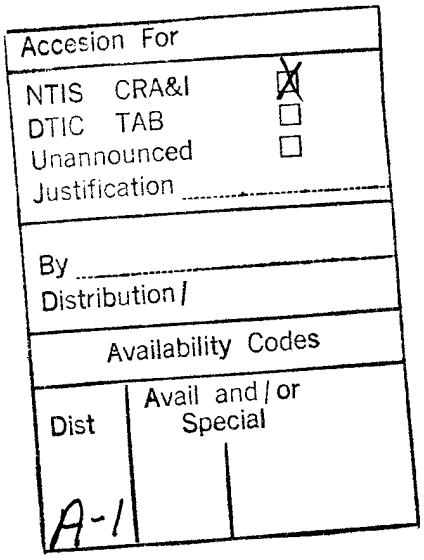
**16. PRICE CODE**

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| Unclassified | Unclassified | Unclassified | UL |

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

architecture will provide the necessary structure needed by Network Managers within DoD to effectively formulate network management strategies that will be able to meet the complex challenges of managing today's information infrastructure.



Approved for public release; distribution is unlimited.

# PROPOSED MANAGEMENT ARCHITECTURE FOR MANAGING A DISTRIBUTED NETWORK INFORMATION SYSTEM

Michael A. Everingham
Lieutenant Commander, United States Navy
B.S., Miami University, 1983

Submitted in partial fulfillment
of the requirements for the degree of

**MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT**

from the

**NAVAL POSTGRADUATE SCHOOL**
**March 1995**

Author: [redacted]
Michael A. Everingham

Approved by: [redacted]
Myung Suh, Principal Advisor

[redacted]
William B. Short, Associate Advisor

[redacted]
David R. Whipple, Chairman
Department of Systems Management

# ABSTRACT

Today's organizations are becoming increasingly dependent on their information infrastructure. Whether it is an effort to maintain their competitive advantage, or provide a Joint Task Force Commander with a real-time picture of the battlefield, the organization's information infrastructure is becoming a vital key to the organization's success or failure in performing its mission-critical processes. "However, the complexity of the modern infrastructure, and the changes affecting it, produce significant management challenges." [Ref. 1, p. 451]

The purpose of this study is to develop a management architecture that will assist Network Managers in formulating network management strategies that will be flexible, proactive, distributed, standardized, comprehensive, collaborative and integratable. It also lays the foundation for the development of an Management Information Base (MIB), specifically that portion which relates to Asset Management.

The need for a comprehensive network management architecture is clear. The proposed architecture will provide the necessary structure needed by Network Managers within DoD to effectively formulate network management strategies that will be able to meet the complex challenges of managing today's information infrastructure.

# TABLE OF CONTENTS

# I. INTRODUCTION

In today's society organizations are becoming increasingly dependent on their information infrastructure. Whether it is an effort to maintain their competitive advantage or provide a Joint Task Force Commander with a real-time picture of the battlefield, the organization's information infrastructure is becoming a vital key to the organization's success or failure in performing its mission-critical processes. "However, the complexity of the modern infrastructure, and the changes affecting it, produce significant management challenges." [Ref. 1, p. 451] Organizations must shift the management of this complex infrastructure from people to computer systems and assist Network Managers in managing those computer systems by designing management architectures which will allow them to more effectively use information and insulate people from the underlying complexities of the organization's information infrastructure.

This study focuses on the development of just such an architecture. The architecture proposed in this study is designed to assist Network Managers in forming management strategies that will allow them to maximize staff effectiveness, effectively distribute management responsibilities, proactively solve network problems, and easily integrate legacy and emerging technologies. Chapter I of this study defines network management and points out some of the driving forces behind the need for organizations to increase their emphasis on network management. Chapter II describes a management architecture that will give Network Managers a general framework in which to formulate their overall network management strategy. This architecture, dubbed DNIMA, or Distributed Network Information Management Architecture, is designed to encompass the commonalties of the various management architectures as well as incorporate ideas from other management philosophies to try to overcome the weaknesses of the individual architectures. However, the details of how the various components of the proposed architecture should communicate or why one way would be preferred to another is beyond the scope of this study. The final chapter of this study presents future research directions based on this study and summarizes the conclusions drawn from this research.

## A. NETWORK MANAGEMENT -- INCREASED EMPHASIS

Networks have grown in size and complexity at an exponential rate. The trend to replace a few centralized mainframes with many geographically distributed client-server systems has caused an explosion in the type and number of components which need to be managed. According to Infonetics Research, Inc. of San Jose, CA, "there has been a 50 percent gain in the number of nodes, 150 percent growth in the number of LANs, and 140 percent increase in the number of hubs, bridges, routers, and other interconnection devices. And as every network manager knows, the challenge of managing a network is magnified exponentially each time a device, protocol, interface, management standard, new traffic load level, or level of service is added." [Ref. 2, p. 84] This network growth will only continue. As military leaders try to coordinate information from all corners of the globe, workers and soldiers become more computer literate, and new technologies continually show up on the marketplace, the managers and systems that are in place to manage these ever growing networks will continually be pushed to their limits.

While organizations continue to invest more dollars into systems and infrastructure, their budgets for managing these resources have not kept pace. In polling 100 Fortune 1000 MIS departments, Infonetics found that while these companies were spending about $650,000 a year on upgrades to the company's infrastructure they were only spending about $60,000 on maintaining and managing that infrastructure. [Ref. 2, p. 84] Due to this gap network managers must rethink and rebuild their management strategies to allow them to perform their responsibilities in a more efficient, timely and proactive manner.

The need for a comprehensive management strategy is further compounded by the lack of standardization. In this world of multiple and often times bewildering array of communication protocols (e.g., SNA, CMIP, and SNMP) and platforms (e.g., Sun, OS/2, Windows, and MVS) network managers must strive to formulate a management strategy to manage this multitude of dissimilar networks connected together. The problem is that today's evolving networks are too diverse, the bandwidth and other conflicts too broad, and the technologies change too quickly for any one standard to be created in the near

future. Even if one could create a standard, the network manager must still contend with a tremendous number of legacy systems that would have to be integrated.

On top of trying to manage an ever expanding heterogeneous network with fewer dollars and little or no standardization, Network Managers have to divvy up the management chores of managing these complex computer networks among fewer and fewer individuals. Surveys conducted by Infonetics indicate there is typically only one network manager for every 300 users. [Ref. 2, p. 84] However, few organizations are able or willing to increase support staffs because of the limited effectiveness of fragmented management methods and probable marginal yield in improved results. [Ref. 1, p. 452]

As our organization's information infrastructure continues to expand at an exponential rate and the budgets and staff to manage it continue to lag behind, managers must depend more and more on network management tools to take up the slack. However, these tools are often made by different vendors, using different architectures with different protocols. Manager must be able to seamlessly integrate these tools in order to keep the organization running smoothly and have as little impact on the user as possible. To make matters worse, Network Managers can not shut down the business in order to incorporate a new technology or fine-tune a routing scheme. Making changes to an organization's infrastructure is "a complex task that can be compared to rebuilding an engine while it's running." [Ref. 2, p. 84] In order to combat these shortfalls, Network Managers must develop management strategies which allow them to maximize staff effectiveness, effectively distribute management responsibilities, proactively solve network problems, and easily integrate legacy and emerging technologies. The first step in developing such a strategy is to clearly define all the responsibilities of operating this growing and complex environment.

## B. NETWORK MANAGEMENT -- DEFINITION

When asked to define network management, one usually defines it as a collection of management responsibilities that a Network Manager would commonly perform in a

networking environment. Typical responsibilities include traffic analysis, fault detection and isolation, and software distribution. However, as you poll more and more individuals you would quickly realize that no two people would come up with the same collection of management responsibilities. In addition, the inconsistency in the meaning of the various management responsibilities adds to the vagueness and confusion as to the definition of network management. Currently, there is no widely accepted term to describe the totality of management responsibilities in the enterprise. [Ref. 3, p.118]

One way to try to eliminate the lack of a common definition of network management is to define it by grouping the various management responsibilities of a Network Manager into several broad functional areas. As with trying to find a common meaning to network management, trying to group management responsibilities into a common set of categories is equally difficult. There are almost as many ways to categorize management responsibilities as there are responsibilities. However, all the numerous arbitrary taxonomies for categorizing network management responsibilities are designed around three network management protocols--Common Management Information Protocol (CMIP), Simple Network Management Protocol (SNMP), and IBM's System Network Architecture (SNA) / Advanced Peer to Peer Networking (APPN). [Ref. 4, p. 28] The following paragraphs will introduce three categorizations of network management responsibilities based on the above management architectures. Each of the categorizations are summarized in Table 1 and will be drawn upon later in developing a unified and comprehensive network management architecture.

The first categorization is the International Standards Organization's (ISO) Open Systems Interconnection (OSI) Systems Management architecture, based on CMIP. This architecture categorizes network management responsibilities into five functional areas--Fault Management, Performance Management, Accounting Management, Configuration Management, and Security Management. "Although this functional classification was developed for the OSI environment, it has gained broad acceptance by vendors of both standardized and proprietary network management systems." [Ref. 5, p. 4]

| OSI CMIP | SNA/APPN | TCP/IP SNMP |
|---|---|---|
| Fault Management | Problem Management | Network Management |
| Performance Management | Performance and Accounting Management | Systems Management |
| Accounting Management | Change Management | Software Management |
| Configuration Management | Configuration Management | Data Management |
| Security Management | | Network Administration |
| | | User Management |

Table 1. Management Functional Areas

The second categorization, IBM's SNA/APPN, categorizes network management responsibilities into four functional areas and until just a few years ago provided a common strategy in defining network management. [Ref. 4, p. 28] But with the emergence of SNMP for Transmission Control Protocol/Internet Protocol (TCP/IP) and CMIP for OSI, the definition of network management has become less clearly defined.

The third possible categorization of network management responsibilities is one developed by Mr. Lenny Liebmann, an independent consultant in New Jersey, who specializes in computer networking. This architecture can be implemented around SNMP for TCP/IP and categorizes network management responsibilities into six functional areas. These six areas differ somewhat from the other two primarily because they were developed based more on current management practices and experiences than from a theoretical or technical perspective.

There are numerous other taxonomies for categorizing network management responsibilities but they all have similarities with the three depicted in Table 1. For the purpose of this study, network management responsibilities will be categorized into eight functional areas--Performance Management, Operations Management, Fault Management, Security Management, Asset Management, Accounting Management, Storage Management, and User Management. These eight areas incorporate what are found in many of the categorizations of network management responsibilities as well as several other functional areas, to provide a more comprehensive definition of network management.

However, defining network management is just the first step in developing a management architecture. The next step is to describe the various components which make up the management architecture and how they should interact with one another in order to assist Network Managers in developing management strategies to design, monitor, control and evaluate the four key elements of an organization's information infrastructure--hardware, software, data and users.

# II. THE NETWORK MANAGEMENT ARCHITECTURE

In a world of heterogeneous networks there is a growing need for a system that will allow Network Managers to organize the vast array of multi-vendor multi-protocol management tools and the growing number of management responsibilities in a logical and cohesive manner. Figure 1 shows a three layer hierarchical architecture designed to assist Network Managers in:

- Organizing their management responsibilities;
- Understanding the relationship between different responsibilities as well as the role these responsibilities play in various levels of decision making;
- Knowing which management tools need to be developed and how they should be integrated into their current network environment;
- Redesigning legacy systems to be better utilized and incorporated into a comprehensive management strategy;
- Introducing emerging technologies and management tools into the current architecture; and
- Formulating their overall network management strategy, a strategy which should be flexible, proactive, distributed, standardized, comprehensive, collaborative and integratable.

At the lowest layer of this architecture, the Physical Layer, are the Network Elements (NE). NEs include the physical devices such as routers, bridges, workstations, etc., as well as logical entities such as services, sessions, routers, etc., that constitute the network. Also at this level are the Element Management Systems (EMS). These systems manage specific subsets of NEs.

The next layer of the architecture, the Aggregation Layer, is composed of a collection of distributed Network Management Stations (NMS). These NMSs monitor

and control a group of NEs and/or EMSs. The data provided by the NEs and EMSs are aggregated and used to optimize the NMS segment of the network.

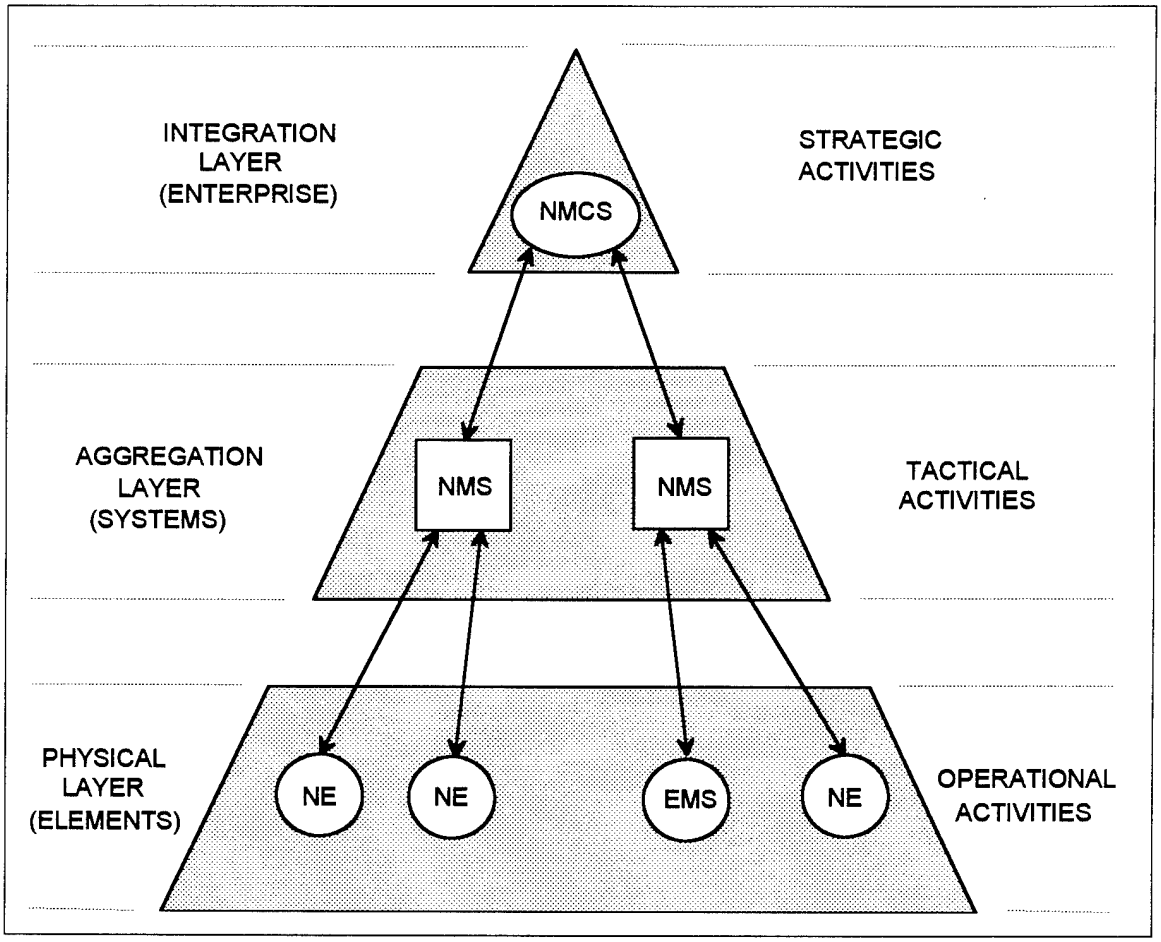



Figure 1. Distributed Network Information Management Architecture (DNIMA)

However, in order to achieve the goals stated previously, the information which has been collected by the NMSs needs to be integrated with the information collected by other NMSs. This is accomplished at the Intergration Layer of the architecture. At this layer information from the NMSs is passed to a central network management control system (NMCS) which is composed of a standard Management Information Base (MIB) and a Management Information Processing System (MIPS). This NMCS provides the

Network Manager with a composite view of the organization's entire networking environment.

Communication between the different layers can be accomplished through the implementation of a standard protocol such as SNMP or through a translation application or gateway that would allow connectivity between components which are made by different vendors using different protocols.

Information that is passed within the hierarchy is grouped into three classes: sensory, structural and controlling. [Ref. 6, p. 87] Sensory data is data received from the network's monitoring processes and includes data such as node queue length, network alarms or security violations. Structual data is data such as CPU type, installed memory or the name of the network operating system. These two types of data generally flow up the hierarchy of the architecture. The third type of data is control data. This data generally flows down the hierarchy and includes variables such as maximum flow on individual trunks and routing tables.

Management functions as well as problem solving are handled at the lowest possible level in the hierarchy. It is much more efficient for a NE or EMS to perform a management function or solve a problem than it is to send information to a higher level component. [Ref. 4, p.26] If information is allowed to indiscriminately flow up to higher levels, it could easily bring the higher level component to its knees due to the sheer volume of information to be managed and analyzed. [Ref. 4, p.26]

The following sections describe each of the three layers of the architecture in more detail and further expands the description of the NMCS.

## A. PHYSICAL LAYER

The Physical Layer, composed of all the physical and logical devices (elements) which constitute the network, handles routine operational tasks. Tasks may include the normal operation of a router, responding to higher level requests for information,

performing basic alarm filtering and correction, and sending higher level configuration or performance information based on pre-established settings.

## B. AGGREGATION LAYER

The second layer of the architecture is called the Aggregation Layer. In this layer a collection of NMSs aggregate the data received from a group of NEs and/or EMSs. This aggregated (system) view of a particular segment or service of the network assists Network Managers in optimizing the performance of a segment and performing other tactical activities. These tactical activities may include fault isolation, automatic fault restoration, or maintenance testing. At this layer there should be heavy emphasis in artificial intelligence (AI). The use of AI can range from automatically re-routing circuits if there is a failure, to alerting operations personnel of impending failure of a network element. Effective use of AI can result in getting problems diagnosed correctly the first time, reducing down time and the resulting impact on the network customer. [Ref. 7, p.427]

By using a distributed management architecture coupled with AI, NMS can more easily collect traps, poll devices, and forward statistics, alarms and status's necessary to proactively monitor the subnet (system). [Ref. 7, p. 429] Pushing some managerial functions down the hierarchy reduces response times, improves performance and allows higher level components more time to perform meaningful trend analysis without impacting the performance of the network.

## C. INTEGRATION LAYER

The last layer of the architecture is the Intergration Layer. This level of the architecture handles the strategic activities required to effectively manage a growing and complex networking environment. "Each of the previously discussed operational and tactical activities result in the measurement and reporting of various network parameters. This collected history is compiled into availability and utilization reports." [Ref. 7, p.433] These and other reports aid in diagnosing chronic fault situations, predicting network

failures and warning managers of potential service impacts, as well as aiding in performing network engineering. The Intergration Layer is composed of a central NMCS which integrates information received from NMSs to provide a comprehensive (enterprise) view of the organization's networking environment.

## 1. Network Management Control System

The NMCS, depicted in Figure 2, has two primary components, the MIB and MIPS. The MIB is the thread that ties the components of the architecture and the management functions together. It is the central depository of all information about an

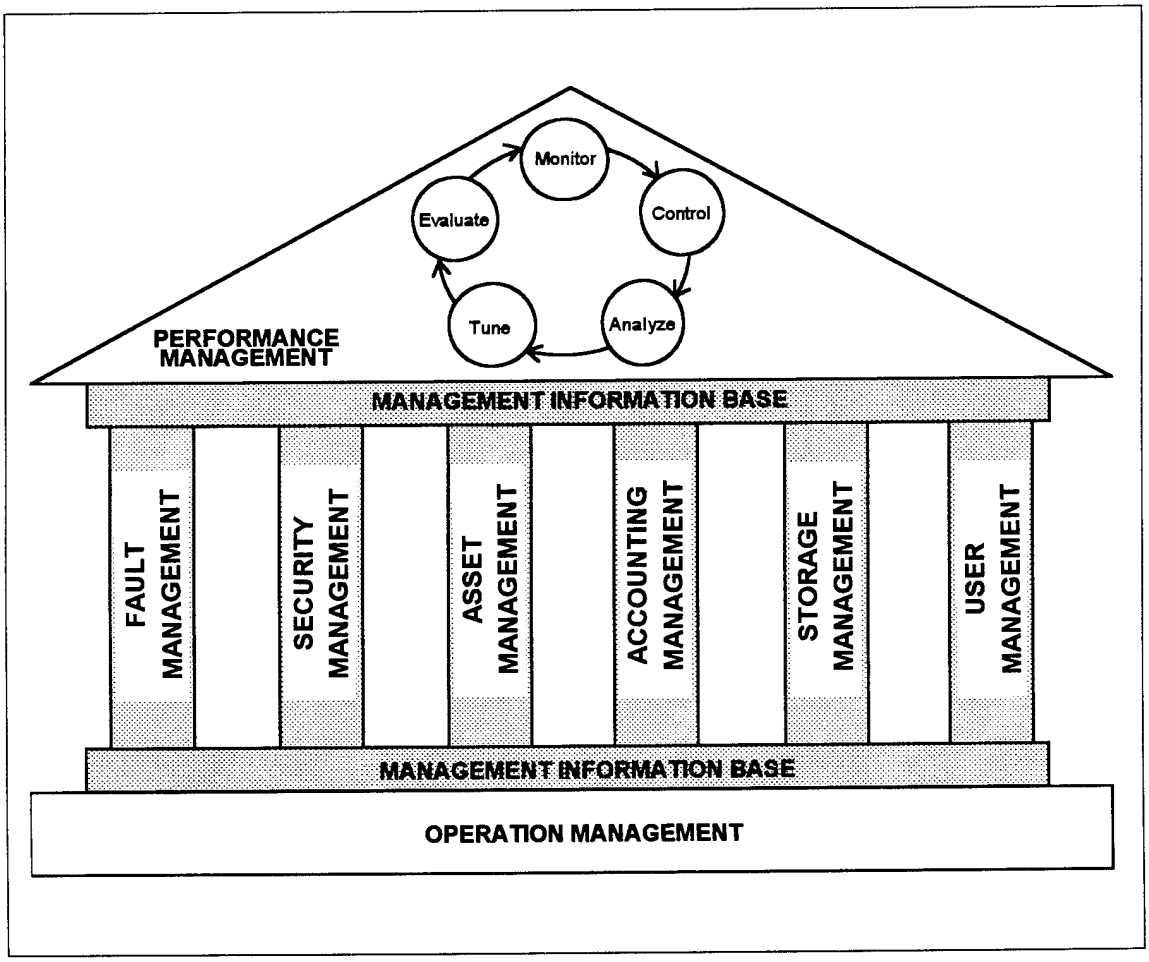



Figure 2. Network Management Control System (NMCS)

organization's information infrastructure. The MIPS is the brain of the architecture and is responsible for coordinating, monitoring, controlling and analyzing the entire information infrastructure.

### *a. Management Information Base*

The MIB is a conceptual repository of information and is the heart of the network management architecture. It provides the interface between all functions of the architecture. "This database is required to store information on network and system configurations, current and historic performance, trouble logs, security codes, accounting information, etc." [Ref. 6, p.85] Logically the MIB is an integrated repository of all network management data but physically is a distributed set of smaller databases that may reside on a NMS, on a NE, or on a central database server at the strategic level. For example, each NE could maintain its own database of descriptive information such as make and model number that when aggregated can provide managers with a detailed inventory picture without the need for physical walk-through inventories. These databases could also be broken down by functions and reside on centralized database servers. For example, one of the primary functions of network management is Asset Management and a key part of that function is maintaining inventory information on NEs and their relationships with one another. Appendices A and B provide the foundation for such a database. Appendix A, the data dictionary, and Appendix B, the entity-relationship diagrams, are designed based on the physical network architecture found at the Fleet Numerical and Oceanographic Command in Monterey, California. [Ref. 8] and [Ref. 9] are also used in the development of this data base structure.

The MIB should exhibit the following characteristics: [Ref. 10, p.152]

- Provide a single logical view but be physically distributed;
- Provide a general naming structure so that the managed objects can be easily stored and retrieved in a uniform way across heterogeneous systems;

- Provide a simple and efficient access interface through which application programs or processes can easily access management information stored in the repository;
- Have an international recognition so that it has the potential for growth and acceptance worldwide;
- Provide a reasonable performance so that the use of it by management tools does not become the bottle neck;
- Provide some form of security in order to regulate the access of important information; and
- Support the replication of management information to increase availability, reliability and performance.

### *b. Management Information Processing System*

The other primary component of the NMCS is the MIPS. The MIPS is merely a collection of management tools that perform the necessary network management tasks in order to effectively control the organization's assets, control the infrastructure's complexity, improve the services provided, assign and control resources to balance various user needs, reduce downtime, increase time between failures, and control costs. [Ref. 5, p.3] These tools could be from a single vendor or from multiple vendors and may perform a single task or function or perform multiple tasks or functions. In addition, these tools could perform tasks that cross functional boundaries. However, in order to more clearly define network management tasks, they are grouped into various functional areas. These network management tasks, as discussed in Chapter I, can be categorized in numerous ways but for the purpose of this study they will be categorized into eight functional areas, each having various sub-functions (Table 2).

Operations Management provides the foundation for the rest of the functional areas in the MIPS and should allow the integration of multi-vendor management tools in a consistent and technology-independent way. This area acts as an

operating system and assists in coordinating the activities of the other seven functional areas to work together as a comprehensive network management solution.

| Functions | Sub-Functions |
|---|---|
| Operation Management | MIB Management, EMS Management, Print & Queue Management, E-mail Management, Remote Monitoring (RMON) Data Display, Scheduling (e.g., backups and software distribution), Report Generation, and User Interfaces |
| Performance Management | Monitoring (i.e., traffic, server, workload, throughput, and LAN/WAN), Analyzing (i.e., traffic, trend, and protocol), Optimization, and Utilization |
| Asset Management | Inventoring, Configuration Management (e.g., automatic detection, mapping and modeling), Capacity Planning, Change Control, Network Design, Cable Plant Design, Software Management (e.g., metering, distribution, licensing, and auditing) |
| Fault Management | Help Desk, Maintenance/Service Information Management, Fault Detection and Isolation, Trouble Ticket Maintenance (e.g., issuing, tracking, and archiving), Trouble Shooting (i.e., local and remote), Auto-Diagnostic, Alarm Management (e.g., setting and notification), Cable Integrity, and Damage Assessment |
| Security Management | Virus Control (i.e., detection, identification/removal, prevention), Access Control (i.e., prevention, authentication, auditing, detection and identification), Encyrption (Key Management), Risk Assesment, Contingency Planning, and Back-Up & Recovery |
| Account Management | Accounting/Billing, Purchasing, Financial Information Management, Charge-Back, and Budgeting |
| Storage Management | Archiving, Library Management (i.e., tape and optical disk), Data Migration, and Capacity Management |
| User Management | Account Management, Customer Support and Training |

Table 2. Network Management Functions and Sub-Functions

Performance Management is the other key functional area of the MIPS. All other functional areas, to varying degrees, contribute to the performance of the organization's information infrastructure. Within this area all tasks revolve around five aspects: [Ref. 11, p.109]

- Monitoring: tracking communication activities in order to gather the appropriate data to evaluate performance;
- Controlling: setting and modifying parameters that control the measurement of performance related data (e.g., threshold values to trigger alerts) in remote systems;
- Analysing: evaluating the results of the measurements in order to obtain assessments of performance;
- Tuning: adjusting resources to improve performance (e.g., modification of resource configuration parameters or redistribution of network traffic);
- Evaluating: comparing performance measures to determine the effectiveness of corrective or preventive actions.

Asset Management is concerned with identifying, monitoring, and controlling an organization's assets (i.e., hardware, software, data and people). Data and people, however, are primarily managed under the functional areas of Storage Management and User Management, respectfully. However, as with these functional areas, there is considerable overlap and interaction between all the various management areas. "For example, Fault Management and Performance Management are closely interrelated, since poor performance is often the only visible symptom of a fault deep down in the system." [Ref. 6, p.86] The MIB is the means by which various functional areas coordinate their activities.

Fault Management is concerned with the detection, isolation, and correction of abnormal operation of the organization's information infrastructure. Fault

management means everything from detection of excessive packet collisions on the network backbone to determining if a printer is jammed. Next to Performance Management there is no quicker way to put a Network Manager in the hot seat than poor Fault Management. Most end-users will tolerate occasional outages but when these outages become too frequent or long in duration their tolerance will end.

Security Management is concerned with providing services and reporting functions that support the implementation of the organization's security policies and protect network resources and user information. Often an after thought in network planning, network security management needs to become an integral part of a network's design. Issues from password management to firewall implementation need to be brought to the forefront.

Accounting Management is concerned with tracking network resource utilization and establishing chargeback mechanisms that can recover the cost of using these resources. The Network Manager needs to be able to track these resources by user or user group in order to :

- Determine if a user or group of users are abusing their access and burdening the network,
- Determine if the resources are being used inefficiently so policies and/or procedures can be modified to improve performance, and
- Be in a better position to plan for network growth.

Storage Management is concerned with the storage, retrieval and security of data. The challenge is balancing the use of magnetic disks with the use of optical disks and capitalizing on the advantages of both technologies. The other key issue is the migration of data from one medium to another (i.e., Which data should be migrated and when should it be migrated?). As the volume and diversity of data generated in distributed

environments increase, effective management of data across the enterprise is that much more imperative. [Ref. 3, p. 119]

The last network management functional area that will be discussed, and often the most overlooked, is User Management. The importance of this area can not be over emphasized. After all, users are the ones that drive the shape of the network and the services that are provided. This area is designed to support the user, from establishing an account on a mainframe to showing them a more efficient way to use a network service. (e.g., E-Mail). However, the word user goes beyond the end-user who sits down at a desk and surfs the World Wide Web (WWW), it goes to the individuals who run and maintain the organization's information infrastructure. Keeping these users trained and proficient at their jobs is as important as ensuring the user on the third floor or across the ocean is able to do his/her job.

## D. COMMUNICATIONS

This really is at the heart of the architecture. How the pieces of the network architecture talk to each other drives how well the goals of this architecture are achieved. If there was a universally accepted protocol for devices to communicate to management consoles and management consoles to talk to other management consoles our problems would just about be solved. Applications, which are developed by different vendors to assist Network Managers in managing this complex array of responsibilities, also need to be able to communicate with the devices and management consoles which make up the network. Providing a common protocol for communication between all the pieces of a network management architecture allows all the pieces to be easily integrated into existing architectures to form a unified and consistent management strategy. It also allows the flexibility for new advances in technology to be more easily incorporated into the current architecture. However, as we all know, there is no end-all be-all standard.

Of the numerous communication standards that are currently on the market, SNMP and CMIP are the most popular. SNMP's strength lies in its simplicity and wide

popularity, whereas CMIP's strength is its functionality and the fact that it is an international standard. Should or could one of these two leading communication protocols overcome its weaknesses and become the end-all be-all standard or will another protocol emerge and become the standard that will solve all our problems? There are many things to consider when choosing the protocols, platforms and applications that will be used in the architecture, but a more detailed discussion of these considerations is beyond the scope of this study.

# III. CONCLUSIONS AND RECOMMENDATIONS

## A. CONCLUSIONS

The purpose of this study was to present a network management architecture which Network Managers could use to assist them in formulating their own network management strategies that would best suit their networking environment. This management strategy should allow them to maximize staff effectiveness, effectively distribute management responsibilities, proactively solve network problems, and easily integrate legacy and emerging technologies. However, there are several problem areas that will challenge the use and implementation of the proposed architecture:

- Standardization: The lack of a universally accepted communication protocol, MIB structure, and terminology will continue to slow the development of a network management system that will allow Network Managers to provide the services the users need so they can effectively perform the organization's key processes. The success of the organization depends on how well these key processes are performed, including the management of the information infrastructure. [Ref. 1, p.461]
- Management tools: As our organization's information infrastructure continues to expand at an exponential rate and the budgets and staff to manage it continue to lag behind, managers must depend more and more on network management tools to take up the slack. However, these tools are often reactive rather than proactive, can not easily be integrated with tools made by other vendors, do not facilitate making timely decisions, or have not yet been developed.
- Changing Environment: Networks have grown in size and complexity at an exponential rate. The trend to replace a few centralized mainframes with many geographically distributed client-server systems has caused an explosion in the type and number of components which need to be managed. This problem is

magnified by economic and regulatory constraints, technology advances, standards development, new product introductions, market requirements, user demands and other factors which change unpredictably over time. [Ref. 12, p.5]

The need for a comprehensive network management architecture is clear. The proposed architecture will provide structure to how an organization's information infrastructure is viewed so Network Managers for both commercial and DoD organizations can effectively formulate a network management strategy that will be flexible, proactive, distributed, standardized, comprehensive, collaborative and integratable.

## B. RECOMMENDATIONS FOR FURTHER STUDY

This study presented many areas that would lend themselves to further study. Since the primary focus of the study was to present a management architecture, it did not go into great detail as to how the various components of the architecture should communicate with one another. It also did not fully detail each of the eight proposed network management functions. Both of these areas would lend themselves to further study and would expand and enhance the proposed architecture and allow a clearer definition of what is meant by network management. Another area for further study would be the continued development of the MIB to encompass other areas of network management besides the portion that was presented in Appendices A and B.

# APPENDIX A. DATA DICTIONARY

The data dictionary includes the semantic and symtactic definition of data elements contained in the various data tables. The descriptions include field name, data type and length, and field description. It also shows the relationship one table has with another. Field name(s) which are underlined denote a key field and those with an asterisk next to them denote a foreign key.

## ADMINISTRATOR

The Administrator table contains information about network administrators.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a network administrator. |
| People_UID* | Num(10) | Identifies a person who is a network administrator. Relates to *UID* in the People table. |

## ALTERNATE FUNCTION

The Alternate Function table contains information about an alternate function a PC or File Server might perform.

| Field Name | Type | Description |
|---|---|---|
| Function_Type | Char(25) | Alternate function a PC or File Server may perform (e.g. Tape Server). |
| PC_UID* | Num(10) | Identifies a PC which is performing an alternate function. Relates to *UID* in the Stand Alone PC table. |
| **or** | | |
| File_Server_UID* | Num(10) | Identifies a File Server which is performing an alternate function. Relates to *UID* in the File Server table. |

## APPLICATION GATEWAY

The Application Gateway table contains information about an application gateway which is a translator between two systems that do not use the same communication protocols, data formatting structures, languages, and/or architecture..

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of an application gateway. |
| PC_UID* | Num(10) | Identifies a PC functioning as an application gateway. Relates to *UID* in the Stand Alone PC table. |
| **or** | | |
| File_Server_UID* | Num(10) | Identifies a file server functioning as an application gateway. Relates to *UID* in the File Server table. |
| Function | Char(10) | Function the application gateway performs (e.g. E-mail). |

## APPLICATION SOFTWARE

The Application Software table contains information about the application software which is installed on a PC or File Server.

| Field Name | Type | Description |
|---|---|---|
| Serial_Number | Num(10) | Unique identifier for a particular copy of an application software. |
| File_Server_UID* | Num(10) | Identifies a file server where an application is installed. Relates to *UID* in the File Server table. |
| **or** | | |
| PC_UID* | Num(10) | Identifies the PC where an application is installed. Relates to *UID* in the Stand Alone PC table. |
| Software_Info_UID* | Num(10) | Identifies the common software information associated with an application. Relates to *UID* in the Software Information table. |

## BUILDING

The Building table contains information about the buildings where the network components are located.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a building. |
| Abbreviation | Char(10) | Common identifier used for a building (e.g. 702). |
| Name | Char(25) | Name of the building. |
| Street | Char(25) | Address of the building. |
| City | Char(20) | City where the building is located. |
| State | Char(2) | State where the building is located. |
| Zip | Char(10) | Zip code of the city where the building is located. |
| Notes | Memo | Any additional information about the building that may be useful. |

## CABLE TYPES

The Cable Types table contains information about the types of cable used in the network.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a type of cable. |
| Type | Char(15) | Type of cable. |
| Company_UID* | Num(10) | Identifies the company which makes the type of cable. Relates to *UID* in the Company table. |
| Number_Conductors | Num(3) | Number of conductors in the type of cable. |
| Label | Char(25) | Common label for the cable type. |
| Part_Number | Char(25) | Part number of the the cable type. |
| Conductor_Type | Char(25) | Type of conductor in the cable. |
| Weight_Per_Length | Num(7.2) | Weight of cable per standard length of cable. |
| Diameter | Num(7.2) | Diameter of the cable in cm. |
| Sheath_Name | Char(10) | Common name for the sheath used in the cable. |

| | | |
|---|---|---|
| UL_Type_Rating | Char(10) | UL type rating of the cable. |
| Trade_Number | Char(25) | Trade number of the cable. |
| Total_Pairs | Num(2) | Number of wire pairs in the cable. |
| Gage | Num(2) | Gage of the cable. |
| Core_Spec | Char(25) | Characteristics of the cable's core. |
| Cladding_Spec | Char(25) | Characteristics of the cable's cladding. |
| Plenum_Flag | Logical | Indicates if the cable has plenum (.T.). |
| Shielded_Flag | Logical | Indicates if the cable is overall shielded (.T.). |

**CARRIER**

The Carrier table contains information about the commercial carrier of a circuit.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a commercial carrier. |
| Name | Char(10) | Name of the commercial carrier (e.g. US SPRINT). |
| Phone | Char(12) | Phone number for the trouble desk of a commercial carrier. |
| Line_Number_Leg_1 | Char(30) | Commercial carrier line identification number for the first leg to the distant end. |
| Line_Number_Leg_2 | Char(30) | Commercial carrier line identification number for the second leg to the distant end. |

**CCO**

The CCO table contains information about the CCO of a circuit.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a CCO. |
| Name | Char(10) | Name of a CCO (e.g. Monterey). |
| Phone | Char(12) | Phone number for a CCO. |

## CD ROM DISK

The CD ROM Disk table contains information about a CD ROM disk which is installed on a CD ROM drive.

| Field Name | Type | Description |
|---|---|---|
| Platter_Number | Num(2) | The number of the platter in a juke box or 1 if the drive is a single disk drive. |
| CD_Drive_UID* | Num(15) | Identifies a CD ROM drive where a disk is installed. Relates to *UID* in the CD ROM Drive table. |
| Software_Info_UID* | Num(10) | Identifies the common software information associated with a CD disk. Relates to *UID* in the Software Information table. |

## CD ROM DRIVE

The CD ROM Drive table contains information about a CD ROM drive.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a CD ROM drive. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a CD ROM drive. Relates to *UID* in the Device Information table. |
| Access_Time | Num(4) | Access time in ms. |
| Transfer_Rate | Num(4) | Transfer rate in KB/sec. |
| CD_ROM_Server_UID* | Num(10) | Identifies a CD ROM server which is associated with a CD ROM drive. Relates to *UID* in the CD ROM Server table. |
| Port_UID* | Num(10) | Identifies the port on a device which is associated with a CD ROM drive. Relates to *UID* in the Port table. |

## CD ROM SERVER

The CD ROM Server table contains information about a CD ROM server.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a CD ROM server. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a CD ROM server. Relates to *UID* in the Device Information table. |

## CIRCUIT

The Circuit table contains information about the communication lines to outside organizations.

| Field Name | Type | Description |
|---|---|---|
| CCSD | Char(4) | Unique identifier of a circuit. |
| Pair_Number | Char(2) | The pair number of a circuit. |
| Circuit_Number | Char(16) | The circuit number. |
| Pacbell_Number | Char(16) | The commercial line number associated with a circuit. |
| Nomenclature | Char(25) | Common name used to describe a circuit. |
| Distant_End | Char(25) | Name of the organization where the line terminates. |
| Security_Level | Char(1) | The security level of the circuit (T - Top Secret, S - Secret, or U - Unclassified). |
| RM_Duties | Char(25) | The RM duties for a circuit (CARP - Contingency Alternate Routing Program, QC - Quality Assurance Checks, TBS - Trouble Shooting, and/or KC - Key Change). |
| OTAR | Char(1) | Identifies the OPTAR (D, H or Blank). |

| | | |
|---|---|---|
| Crypto_Load | Char(1) | The crypto load times for the circuit (D - Daily, M - Monthly, Q - Quarterly, A - Annually, or N - No Load). |
| Backup_CCSD | Char(4) | Identifies the backup circuit. |
| Baud_Rate | Char(4) | Baud rate of a circuit in KB/sec. |
| Crypto | Char(6) | Type of crypto used. |
| Modem | Char(10) | Type of modem used. |
| CCO_UID* | Num(10) | Identifies the CCO of a circuit. Relates to *UID* in the CCO table. |
| Carrier_UID* | Num(10) | Identifies the commercial carrier of a circuit. Relates to *UID* in the Carrier table. |
| Terminal_Code* | Char(1) | Identifies the terminal block associated with a circuit. Relates to *Block_Code* in the Terminal Block table. |
| Host_Name* | Char(10) | Identifies the host attached to a circuit. Relates to *Host_Name* in the Host table. |
| Supernet_UID* | Num(10) | Identifies the supernet associated with a circuit. Relates to *UID* in the Supernet table. |

**CIRCUIT / USER GROUP RELATION**

The Circuit / User Group Relation table provides the link for a many-to-many relationship between the Circuit table and the User Group table.

| Field Name | Type | Description |
|---|---|---|
| User_Group_UID* | Num(10) | Identifies a user group which is associated with a circuit. Relates to *UID* in the User Group table. |
| Circuit_CCSD * | Char(4) | Identifies a circuit which is associated with a user group. Relates to *CCSD* in the Circuit table. |

## COMMUNICATION SERVER

The Communication Server table contains information about a communication server.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a communication server. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a communication server. Relates to *UID* in the Device Information table. |

## COMPANY

The Company table contains information about the companies which 1) make various components used in the network, 2) provide maintenance or warranty services and 3) sell various components used in the network.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a company. |
| Abbreviation | Char(10) | Common abbreviation used for a company name. |
| Name | Char(25) | Name of a company. |
| Street | Char(25) | Address of a company. |
| City | Char(20) | City where a company is located. |
| State | Char(2) | State where a company is located. |
| Zip | Char(10) | Zip code of the city where a company is located. |
| Phone_Number | Char(12) | Phone number of the company. |
| Manufacturer_Flag | Logical | Indicates if the information is for a company who makes network components (.T.). |
| Vendor_Maint_Flag | Logical | Indicates if the information is for a company who provides maintenance services (.T.). |
| Vendor_Warranty_Flag | Logical | Indicates if the information is for a company who provides warranty services (.T.). |

| | | |
|---|---|---|
| Distributer_Flag | Logical | Indicates if the information is for a company who sells network components (.T.). |
| Notes | Memo | Any additional information about a company that may be useful. |

## CONCENTRATOR

The Concentrator table contains information about a concentrator which centrally locates the wire attachments from various workstations.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a concentrator. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a concentrator. Relates to *UID* in the Device Information table. |

## CONDUCTOR CATEGORY

The Conductor Category table contains information about a category of conductor.

| Field Name | Type | Description |
|---|---|---|
| Category | Char(25) | Unique identifier of a conductor category. |
| Conductor_Group* | Char(15) | Identifies a group assigned to a conductor category. Relates to *Group* in the Conductor Group table. |

## CONDUCTOR CATEGORY / CONNECTOR VALUE RELATION

The Conductor Category / Connector Value Relation table provides the link for a many-to-many relationship between the Conductor Category table and the Connector Value table.

| Field Name | Type | Description |
|---|---|---|
| Conductor_Cat* | Char(25) | Identifies a conductor category which is associated with a connector value. Relates to *Category* in the Conductor Cats table. |
| Connector_Value* | Char(15) | Identifies a connector value which is associated with a conductor category. Relates to *Value* in the Connector Values table. |

## CONDUCTOR GROUP

The Conductor Group table contains information about the grouping of conductor categories.

| Field Name | Type | Description |
|---|---|---|
| Group | Char(15) | Unique identifier of a conductor group. |
| Description | Char(64) | Description of a conductor group. |

## CONDUCTOR GROUP / CONNECTOR VALUE RELATION

The Conductor Group / Connector Value Relation provides the link for a many-to-many relationship between the Conductor Group table and the Connector Value table.

| Field Name | Type | Description |
|---|---|---|
| Conductor_Group* | Char(15) | Identifies the conductor group which is associated with a connector value. Relates to *Group* in the Conductor Group table. |

| | | |
|---|---|---|
| Connector_Value* | Char(15) | Identifies the connector value which is associated with a conductor group. Relates to *Value* in the Connector Values table. |

**CONNECTION**

The Connection table contains information on the connections between devices.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a connection between devices. |
| Conn_Type_UID* | Num(10) | Identifies a connector type which is associated with a connection. Relates to *UID* in the Connector Type table. |
| Phy_Cable_UID* | Num(10) | Identifies the physical cable which connects the two devices. Relates to *UID* in the Physical Cable table. |

**CONNECTOR TYPE**

The Connector Type table contains information about the types of connectors used in the network.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a type of connector. |
| Type | Char(25) | Common name of a connector type. |
| Max_Pins | Num(2) | Maximum number of pins associated with a connector type. |
| Gender_Abbr* | Char(15) | Identifies the gender abbreviation of the connector. Relates to *Abbreviation* in the Gender table. |
| Conductor_Cat* | Char(15) | Identifies the category of conductor associated with a type of connector. Relates to *Category* in the Conductor Cats table. |
| Part_Number | Char(25) | Part Number for a connector type. |

| | | |
|---|---|---|
| Company_UID* | Num(10) | Identifies the company which makes a connector type. Relates to *UID* in the Company table. |
| Description | Char(64) | Description of the connector type. |

**CONNECTOR VALUE**

The Connector Value table contains information about a connector used in the network.

| Field Name | Type | Description |
|---|---|---|
| Value | Char(15) | Unique identifier for a value of a connector. |
| Description | Char(64) | Description of a connector value. |

**DEVICE INFORMATION**

The Device Information table contains information which is common to all network components.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a network component's common information. |
| Company_UID* | Char(20) | Identifies the company which makes a network component. Relates to *UID* in the Company table. |
| Model_Number | Char(15) | Model number of a network component. |
| Serial_Number | Char(15) | Serial number of a network component. |
| Installed_Date | Date | Date a network component is installed in the network. |
| Receipt_Date | Date | Date a network component is received. |
| Est_Repl_Date | Date | Date a network component is expected to be replaced. Refers to a network component's life cycle. |

| | | |
|---|---|---|
| Maint_Reqn_Num* | Char(16) | Identifies the maintenance contract to which a network component is covered. Relates to *requisition number* in the Maintenance Contract table. |
| Purchase_Reqn_Num* | Char(16) | Identifies the purchase order under which a network component was procured. Relates to *requisition number* in the Purchase table. |
| Space_UID* | Num(10) | Identifies the space where a network component is located. Relates to *UID* in the Space table. |
| Warranty_Number* | Char(25) | Identifies the warranty under which a network component is covered. Relates to *warranty number* in the Warranty table. |
| Notes | Memo | Any additional information about a network component that may be useful. |

**DEVICE TYPE**

The Device Type table contains information about a device type.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a device type. |
| Type | Char(25) | Common name of a device type (e.g. Zenith 248). |
| Notes | Memo | Any additional information about a device type that may be useful. |

**DEVICE TYPE / PARTS RELATION**

The Device Type / Parts Relation table provides the link for a many-to-many relationship between the Device Type table and the Parts table.

| Field Name | Type | Description |
|---|---|---|
| Parts_UID* | Num(10) | Identifies a spare part that can be used to repair a particular device type. Relates to *UID* in Parts table. |

| | | |
|---|---|---|
| Device_Type_UID* | Num(10) | Identifies a device type which a spare part can be used to repair. Relates to *UID* in the Device Type table. |

**DISK**

The Disk table contains information about the hard disks associated with a file server.

| Field Name | Type | Description |
|---|---|---|
| Device_Number | Num(10) | Unique identifier of a disk. |
| Card | Char(15) | Description of the controller card for the disk (e.g. 16-bit ISA). |
| Controller | Char(10) | Type of controller associated with a disk. |
| Type | Num(2) | Type of hard disk installed. |
| Heads | Num(2) | Number of heads. |
| Cylinders | Num(4) | Number of cylinders. |
| Sectors | Num(4) | Number of sectors. |
| Sectors_Per_Track | Num(4) | Number of sectors per track. |
| Capacity | Num(4) | Capacity of hard disk in MB. |
| Disk_System_Num* | Num(10) | Identifies a disk system a disk. Relates to *Disk_System_Number* in the Disk System table. |

**DISK SYSTEM**

The Disk System table contains information about the disk system of a file server.

| Field Name | Type | Description |
|---|---|---|
| Disk_System_Number | Num(10) | Unique identifier of a disk system. |
| Drive_Name | Char(15) | Name of the drive on the file server. |
| IO_Setting | Char(10) | Input/Output settings. |
| Slot_Number | Num(2) | Slot number. |
| File_Server_UID* | Num(10) | Identifies a file server were the disk system is located. Relates to *UID* in the File Server table. |

## DISKETTE DRIVE

The Diskette Drive table contains information about the diskette drive associated with a PC or SUN Terminal.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a diskette drive. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a diskette drive. Relates to *UID* in the Device Information table. |
| Type | Num(2) | Type of diskette drive installed. |
| Heads | Num(2) | Number of heads. |
| Cylinders | Num(4) | Number of cylinders. |
| Sectors | Num(4) | Number of sectors. |
| Capacity | Num(4) | Capacity of diskette drive in KB. |
| PC_UID* | Num(10) | Identifies the PC which is associated with a diskette drive. Relates to *UID* in the Stand Alone PC table. |
| Sun_UID* | Num(10) | Identifies the Sun terminal which is associated with a diskette drive. Relates to *UID* in the SUN Terminal table. |

## DUMB TERMINAL

The Dumb Terminal table contains information about a dumb terminal.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a dumb terminal. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a dump terminal. Relates to *UID* in the Device Information table. |

**FILE SERVER**

The File Server table contains information about a file server.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a file server. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a file server. Relates to *UID* in the Device Information table. |

**GENDER**

The Gender table contains information about the gender of a connector type.

| Field Name | Type | Description |
|---|---|---|
| Abbreviation | Char(15) | Unique identifier of a gender associated with a connector type. |
| Name | Char(25) | Descriptive name of a gender. |

**HARD DISK**

The Hard Disk table contains information about a hard disk associated with a PC.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a hard disk. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a hard disk. Relates to *UID* in the Device Information table. |
| Type | Num(2) | Type of hard disk installed. |
| Heads | Num(2) | Number of heads. |
| Cylinders | Num(4) | Number of cylinders. |
| Sectors | Num(4) | Number of sectors. |
| Capacity | Num(4) | Capacity of a hard disk in MB. |

| | | |
|---|---|---|
| PC_UID* | Num(10) | Identifies the PC which is associated with a hard disk. Relates to *UID* in the Stand Alone PC table. |

**HOST**

The Host table contains information about host computer.

| Field Name | Type | Description |
|---|---|---|
| Host_Name | Num(10) | Unique identifier of a host computer (e.g. DPSR). |
| Classified_Flag | Logical | Indicates if a host computer is classified (.T.). |

**MAINTENANCE CONTRACT**

The Maintenance Contract table contains information about the maintenance services on network components.

| Field Name | Type | Description |
|---|---|---|
| Requisition_Number | Num(16) | Unique identifier of a maintenance contract. |
| Contract_Number | Num(16) | Contract number of the maintenance contract. |
| Start_Date | Date | Date the maintenance contract goes in to effect. |
| End_Date | Date | Date the maintenance contract expires. |
| Type_of_Service | Char(25) | Type of maintenance services which are performed under the contract. |
| Company_UID* | Num(10) | Identifies the company which provides the maintenance services. Relates to *UID* in the Company table. |

## MAINTENANCE CONTRACT / PEOPLE RELATION

The Maintenance Contract / People Relation table provides the link for a many-to-many relationship between the Maintenance Contract table and the People table.

| Field Name | Type | Description |
|---|---|---|
| Maint_Reqn_Num* | Char(16) | Identifies a maintenance contract a person is authorized to call about to obtain maintenance services. Relates to *Requisition_Number* in the Maintenance Contract table. |
| People_UID* | Num(10) | Identifies a person who is authorized to contact a company to obtain maintenance services. Relates to *UID* in the People table. |

## MODEM

The Modem table contains information about a modem which provides the connections for computers into the public switched telephone network (PSTN).

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a modem. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a modem. Relates to *UID* in the Device Information table. |
| Type | Char(3) | Type of modem (external or internal). |
| Fax_Capable_Flag | Logical | Indicates if the modem is capable of sending and receiving faxes (.T.). |
| Data_Speed | Char(5) | Speed which a modem can transfer data in bps (e.g. 14.4). |
| Fax_Speed_Send | Char(5) | Speed which a modem can send a fax in bps (e.g. 9600). |
| Fax_Speed_Rec | Char(5) | Speed which a modem can receive a fax in bps (e.g. 9600). |
| Error_Correction | Char(10) | Type of error correction (e.g. V.42). |
| Compression | Char(5) | Type of compression (e.g. MNP 5). |

| | | |
|---|---|---|
| Port_UID* | Num(10) | Identifies the port on a device which is associated with a modem. Relates to *UID* in the Port table. |

**MONITOR**

The Monitor table contains information about a monitor.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a monitor. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a monitor. Relates to *UID* in the Device Information table. |
| Size | Num(2) | Monitor size in inches. |
| Type | Char(4) | Type of monitor (e.g. SVGA) |
| Max_Resolution | Char(9) | Max resolution of the monitor (e.g. 800 x 600). |
| Max_Scan_Freq | Num(2) | Maximum scan rate in Hz. |
| Min_Scan_Freq | Num(2) | Minimum scan rate in Hz. |
| Max_Refresh_Freq | Num(2) | Maximum refresh rate in Hz. |
| Min_Refresh_Freq | Num(2) | Minimum refresh rate in Hz. |
| Connection_Types | Char(25) | Type of possible connectors (i.e. RGB, VGA - 15 pin, EGA - 9 pin, etc.). |
| Interlaced_Flag | Logical | Indicates if the monitor is interlaced (.T.). |
| PC_UID* | Num(10) | Identifies the PC which is attached to the monitor. Relates to *UID* in the Stand Alone PC table. |
| **or** | | |
| SUN_UID* | Num(10) | Identifies the SUN terminal which is attached to the monitor. Relates to *UID* in the SUN Terminal table. |
| **or** | | |
| XTerminal_UID* | Num(10) | Identifies the XTerminal which is attached to the monitor. Relates to *UID* in the XTerminal table. |
| **or** | | |
| Dumb_UID* | Num(10) | Identifies the dumb terminal which is attached to the monitor. Relates to *UID* in the Dumb Terminal table. |

## MULTIPORT REPEATER

The Multiport Repeater table contains information about a multiport repeater which boosts a signal from multiple devices so that the length of the network can be extended.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a multiport repeater. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a multiport repeater. Relates to *UID* in the Device Information table. |
| Subnet_UID* | Num(10) | Identifies the subnet to which a multiport repeater is a part of. Relates to *UID* in the Subnet table. |
| Max_Slots | Num(2) | Maximum number of slots in a multiport repeater. |
| Available_Slots | Num(2) | Number of slots which are empty. |

## MULTIPORT TRANSCEIVER

The Multiport Tranceiver table contains information about multiport tranceivers which provide the interface between various nodes and the network.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a multiport transceiver. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a multiport transceiver. Relates to *UID* in the Device Information table. |
| Cascaded_With_UID* | Num(10) | Identifies the transceiver which is cascaded with a multiport transceiver. Relates to *UID* in the Transceiver table. |
| Max_Slots | Num(2) | Maximum number of slots in a multiport transceiver. |
| Available_Slots | Num(2) | Number of slots which are empty. |

## NETWORK

The Network table contains information about the organization's networks. One or more networks are connected to form a supernet. A network is made up of one or more subnets.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a network. |
| Network_Name | Char(25) | Network's common name. |
| Topography | Char(10) | Topography of a network (bus, star, ring, etc.). |
| Administrator_UID* | Num(10) | Identifies a network administrator. Relates to *UID* in the Administrator table. |
| Supernet_UID * | Num(10) | Identifies the supernet which is associated with a network. Relates to *UID* in the Supernet table. |
| Router_UID* | Num(10) | Identifies the router associated with a network. Relates to *UID* in the Router table. |

## NETWORK CARD

The Network Card table contains information about a network card which is associated with a PC.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a network card. |
| Interface_Number | Char(15) | Interface number associated with a network card. |
| Interface_Address | Char(15) | Interface address associated with a network card. |
| Symbolic_Name | Char(10) | Common name for a network card. |
| IO_Port_Address | Char(15) | Input/Output port address. |
| DMA_Setting | Char(10) | DMA settings. |
| IRQ_Setting | Num(2) | IRQ settings. |
| RAM_Address | Char(10) | RAM address. |
| LAN_Board_Type | Char(15) | Type of network board. |

| | | |
|---|---|---|
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a network card. Relates to *UID* in the Device Information table. |
| Slot_Number* | Num(2) | Identifies the expansion slot which is associated with a network card. Relates to *Slot_Number* in the Slot table. |

**NETWORK DRIVER SOFTWARE**

The Network Driver Software table contains information about the network software which is installed on a PC.

| Field Name | Type | Description |
|---|---|---|
| Serial_Number | Num(10) | Unique identifier for a particular copy of network software. |
| Name | Char(15) | Name of the network software. |
| LSL_Version | Char(5) | LSL version number of the network software. |
| LAN_Driver_Version | Char(5) | LAN driver version number of the network software. |
| LAN_Driver_Name | Char(25) | Description or file name of the driver (e.g. SMC8000.COM) |
| IPX_Version | Char(5) | IPX version number of the network software. |
| Shell_Version | Char(5) | Shell version number of the network software. |
| Developer | Char(15) | Name of the developer of the network software. |
| PC_UID* | Num(10) | Identifies a PC where the network software is installed. Relates to *UID* in the Stand Alone PC table. |

## OS SOFTWARE

The OS Software table contains information about the OS software which is installed on a PC or Server.

| Field Name | Type | Description |
|---|---|---|
| Serial_Number | Num(10) | Unique identifier for a particular copy of OS software. |
| Software_Info_UID* | Num(10) | Identifies the common software information associated with OS software. Relates to *UID* in the Software Information table. |
| **or** | | |
| File_Server_UID* | Num(10) | Identifies a file server where the OS software is installed. Relates to *UID* in the File Server table. |
| **or** | | |
| PC_UID* | Num(10) | Identifies a PC where the OS software is installed. Relates to *UID* in the Stand Alone PC table. |
| **or** | | |
| Comm_Server_UID* | Num(10) | Identifies a communication server where the OS software is installed. Relates to *UID* in the Communication Server table. |
| **or** | | |
| CD_ROM_Server_UID* | Num(10) | Identifies a CD ROM server where the OS software is installed. Relates to *UID* in the CD ROM Server table. |
| **or** | | |
| Print_Server_UID* | Num(10) | Identifies a print server where the OS software is installed. Relates to *UID* in the Print Server table. |
| **or** | | |
| Tape_Server_UID* | Num(10) | Identifies a tape server where the OS software is installed. Relates to *UID* in the Tape Server table. |

## PARTITION

The Partition table contains information about the partitions on a hard disk which is associated with a file server.

| Field Name | Type | Description |
|---|---|---|
| Partition_Number | Num(10) | Unique identifier of a partition. |
| Type | Char(10) | Type of disk partition. |
| Logical_Partition_Num | Num(2) | Number of the logical partition. |
| Size | Num(4) | Size of the partition in MB. |
| Starting_Block | Num(10) | Starting block of the partition. |
| Total_Blocks | Num(10) | Total number of blocks in the partition. |
| Mirrored_Partition_Num | Num(2) | Number of the partition the partition is being mirrored by. |
| Disk_Number* | Num(10) | Identifies the disk associated with a partition. Relates to *Disk Number* in the Disk table. |

## PARTITION / VOLUME RELATION

The Partition Volume Relation table provides the link for a many-to-many relationship between the Partition table and the Volume table.

| Field Name | Type | Description |
|---|---|---|
| Partition_Number* | Num(10) | Identifies a partition which is associated with a volume. Relates to *Partition Number* in the Partition table. |
| Volume_Number* | Num(10) | Identifies a volume which is associated with a partition. Relates to *Volume Number* in the Volume table. |

## PARTS

The Parts table contains information about the spare parts available to fix network components.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a spare part. |
| Description | Memo | Description of the spare parts. |
| Part_Number | Char(15) | Part number of the spare part. |
| Company_UID* | Num(10) | Identifies the company that makes the spare part. Relates to *UID* in the Company table. |
| Availability_Flag | Logical | Indicates if the part is available (.T.). |
| Space_UID* | Num(10) | Identifies the space where the spare part is located. Relates to *UID* in the Space table. |
| Purchase_Reqn_Num* | Char(16) | Identifies the purchase order under which the part was procured. Relates to *Requisition_Number* in the Purchase table. |

## PATCH PANEL

The Patch Panel table contains information about a patch panel where network twisted-pair cables terminate and allows the network administrator to easily connect, move, test, and disconnect network devices.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a patch panel. |
| Panel_Type_UID* | Num(10) | Identifies a patch panel type. Relates to *UID* in the Patch Panel Type table. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a patch panel. Relates to *UID* in the Device Information table. |
| Installed_Method | Char(25) | Method used to install a patch panel. |

| | | |
|---|---|---|
| Implementation_Status | Char(25) | Implementation status of a patch panel. |
| Function_Status | Char(25) | Functional status of a patch panel. |

**PATCH PANEL TYPE**

The Patch Panel Type table contains information about the types of patch panels.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a patch panel type. |
| Type | Char(15) | Type of patch panel. |
| Default_Num_Jacks | Num(3) | Default number of jacks in a patch panel type. |
| Default_Offset | Char(10) | Default offset used in a patch panel type. |
| Conn_Type_UID* | Num(10) | Identifies a type of connector. Relates to UID in the Connector Type table. |
| Conductor_Jack | Char(10) | Description of the conductor jack used in a patch panel type. |
| Part_Number | Char(25) | Part number of a patch panel type. |
| Company_UID* | Num(10) | Identifies a company which makes a patch panel type. Relates to *UID* in the Company table. |
| Description | Char(25) | Description of the patch panel type. |

**PEOPLE**

The People table contains information about the people who have access to the network, are system adminstrators, authorized callers for maintenance services or are POCs at a company.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a person. |
| Name | Char(25) | Name of the person. |
| Phone_Number | Char(12) | Phone number of a person. |
| Fax_Number | Char(12) | Fax number of a person. |
| Email_Address | Char(25) | E-mail address of a person. |

| | | |
|---|---|---|
| Space_UID* | Num(10) | Identifies the space where a person is located. Relates to *UID* in the Space table. |
| Department_Code | Char(4) | Department code of the person. |
| Administrator_Flag | Logical | Indicates if a person is a system adminstrator (.T.). |
| Service_Caller_Flag | Logical | Indicates if a person is an authorized caller for maintenance services (.T.). |
| Company_POC_Flag | Logical | Indicates if a person is a POC at a company (.T.). |
| Company_UID* | Num(10) | Identifies the company a person is a POC for. Relates to *UID* in the Company table. |

**PHYSICAL CABLE**

The Physical Cable table contains information about the cable between two devices.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a length of cable. |
| Name | Char(25) | Common name used for a cable. |
| Cable_Type_UID* | Num(10) | Identifies a type of cable which connects two devices. Relates to *UID* in the Cable Type table. |
| Date_Installed | Date | Date the cable was laid. |
| Drawing_Number | Char(25) | Drawing number of the detailed drawing showing the cable and connectors. |

**PORT**

The Port table contains information on the connections within a network device.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a device port. |
| Port_Number | Num(2) | Number of the port within a device. |
| Port_Type | Num(10) | Type of port. |
| Port_Name | Char(32) | Common name for a port. |
| Port_Desc | Char(64) | Description of a port. |

| | | |
|---|---|---|
| Hardware_Address | Char(64) | Hardware address of a port. |
| Hardware_Ver | Char(16) | Hardware version associated with a port. |
| Slot_Number* | Num(2) | Identifies a slot which is associated with a port. |
| Router_UID* | Num(10) | Identifies a router which is associated with a port . Relates to *UID* in the Router table. |
| **or** | | |
| CD_ROM_Server_UID* | Num(10) | Identifies a CD ROM server to which is associated with a port . Relates to *UID* in the CD ROM Server table. |
| **or** | | |
| Comm_Server_UID* | Num(10) | Identifies a comm server which is associated with a port . Relates to *UID* in the Communication Server table. |
| **or** | | |
| Concentrator_UID* | Num(10) | Identifies a concentrator which is associated with a port . Relates to *UID* in the Concentrator table. |
| **or** | | |
| Dumb_Terminal_UID* | Num(10) | Identifies a Dumb Terminal which is associated with a port . Relates to *UID* in the Dumb Terminal table. |
| **or** | | |
| File_Server_UID* | Num(10) | Identifies a file server which is associated with a port . Relates to *UID* in the File Server table. |
| **or** | | |
| Multiport_Repeater_UID* | Num(10) | Identifies a multiport repeater which is associated with a port . Relates to *UID* in the Multiport Repeater table. |
| **or** | | |
| Multiport Tranceiver_UID* | Num(10) | Identifies a multiport tranceiver which is associated with a port . Relates to *UID* in the Multiport Tranceiver table. |
| **or** | | |
| Patch Panel_UID* | Num(10) | Identifies a patch panel which is associated with a port . Relates to *UID* in the Patch Panel table. |
| **or** | | |
| Print_Server_UID* | Num(10) | Identifies a print server which is associated with a port . Relates to *UID* in the Print Server table. |
| **or** | | |
| PC_UID* | Num(10) | Identifies a PC to which is associated with a port . Relates to *UID* in the PC table. |
| **or** | | |
| Star Coupler_UID* | Num(10) | Identifies a print server which is associated with a port . Relates to *UID* in the Print_Servertable. |
| **or** | | |

| | | |
|---|---|---|
| Sun_Terminal_UID* | Num(10) | Identifies a sun terminal which is associated with a port . Relates to *UID* in the Sun Terminal table. |
| **or** | | |
| Tape Server_UID* | Num(10) | Identifies a tape server which is associated with a port . Relates to *UID* in the Tape Server table. |
| **or** | | |
| Terminal Server_UID* | Num(10) | Identifies a terminal server which is associated with a port . Relates to *UID* in the Terminal Server table. |
| **or** | | |
| Tranceiver_UID* | Num(10) | Identifies a tranceiver which is associated with a port . Relates to *UID* in the Tranceiver table. |
| **or** | | |
| XTerminal_UID* | Num(10) | Identifies a XTerminal which is associated with a port . Relates to *UID* in the XTerminal table. |

**PORT ADDRESS**

The Port Address table contains information on the port addresses for each port connection associated with a network device.

| Field Name | Type | Description |
|---|---|---|
| Address | Char(64) | Unique identifier of a port address. |
| Protocol_UID* | Num(10) | Identifies a protocol associated with a port address. Relates to *UID* in the Protocol table. |
| Port_UID* | Num(10) | Identifies a port associated with a network device. Relates to *UID* in the Port table. |
| Address_Type | Binary | Identifies the type of address (1000 - IP address, 0100 - IPX address, 0010 - Node Address, 0001 - Network address). |

## PRINT DRIVER

The Print Driver table contains information about the print drivers associated with a printer.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a print driver. |
| Name | Char(25) | Descriptive name or file name of a print driver. |

## PRINT SERVER / FILE SERVER RELATION

The Printer Server / File Server Relation table provides the link for a many-to-many relationship between the Print Server table and the File Server table.

| Field Name | Type | Description |
|---|---|---|
| Print_Server_UID* | Num(10) | Identifies a print server which is associated with a file server. Relates to *UID* in the Print Server table. |
| File_Server_UID* | Num(10) | Identifies a file server which is associated with a print server. Relates to *UID* in the File Server table. |

## PRINT QUEUE

The Print Queue table contains information about the print queues which are located on a file server.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a print queue. |
| File_Server_UID* | Num(10) | Identifies a file server where a print queue is located. Relates to *UID* in the File Server table. |

**PRINT SERVER**

The Print Server table contains information about a print server which provides user access to printers attached to the network and manages print jobs through a print queue system..

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a print server. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a print server. Relates to *UID* in the Device Information table. |

**PRINTER**

The Printer table contains information about a printer.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a printer. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a printer. Relates to *UID* in the Device Information table. |
| Print_Server_Flag | Logical | Indicates if a printer is performing the function of a print server (.T.). |
| Min_Resolution | Num(4) | Min resolution in dpi. |
| Max_Resolution | Num(4) | Max resolution in dpi. |
| Type | Char(10) | Type of printer (e.g. Ink jet). |
| Colors | Num(8) | Number of colors printer supports. |
| PS_Capable_Flag | Logical | Indicates if a printer is PS capable. |
| Print_Speed_Draft | Num(3) | Speed of a printer in draft mode in cps or ppm, depending on the type of printer. |
| Print_Speed_LQ | Num(3) | Speed of a printer in Letter Quality (LQ) mode in cps or ppm, depending on the type of printer. |
| Buffer_Installed | Num(4) | Size of the installed buffer in KB. |
| Max_Buffer | Num(4) | Max size of the buffer in KB. |

| | | |
|---|---|---|
| Print_Server_UID* | Num(10) | Indentifies the print server associated with a printer. Relates to *UID* in the File Server table. |
| Port_UID* | Num(10) | Identifies the port on a device which is associated with a printer. Relates to *UID* in the Port table. |

**PRINTER / PRINT DRIVER RELATION**

The Printer / Print Driver Relation table provides the link for a many-to-many relationship between the Printer table and the Print Driver table.

| Field Name | Type | Description |
|---|---|---|
| Printer_UID* | Num(10) | Identifies a printer which is associated with a print driver. Relates to *UID* in the Printer table. |
| Print_Driver_UID* | Num(10) | Identifies a print driver which is associated with a printer. Relates to *UID* in the Print Driver table. |

**PRINTER / PRINT QUEUE RELATION**

The Printer / Print Queue Relation table provides the link for a many-to-many relationship between the Printer table and the Print Queue table.

| Field Name | Type | Description |
|---|---|---|
| Printer_UID* | Num(10) | Identifies a printer which is associated with a print queue. Relates to *UID* in the Printer table. |
| Print_Queue_UID* | Num(10) | Identifies a print queue which is associated with a printer. Relates to *UID* in the Print Queue table. |

## PROTOCOL

The Protocol table contains information on a protocol.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a protocol. |
| Type_Desc | Char(64) | Description of a protocol. |

## PURCHASE

The Purchase table contains procurement information about network components.

| Field Name | Type | Description |
|---|---|---|
| Requisition_Number | Num(16) | Unique identifier of a procurement action. |
| Purchase_Order_Number | Num(16) | Purchase order number of a procurement action. |
| Line_Item_Number | Num(3) | Line number of a network component on a purchase order. |
| Line_Item_Price | Num(9.2) | Price of a network component. |
| Purchase_Date | Date | Date a network component was purchased. |
| Company_UID* | Num(10) | Identifies the company a network component was procured from. Relates to *UID* in the Company table. |

## ROUTER

The Router table contains information about a router that interconnects networks over local or wide areas and provides traffic control and filtering functions when more than one pathway exists between two end-points on the network.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a router. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a router. |

| | | |
|---|---|---|
| | | Relates to *UID* in the Device Information table. |
| Software_Desc | Char(32) | Description of the configuration software. |
| Software_Ver | Char(16) | Version of the configuration software. |
| Hardware_Desc | Char(32) | Description of the configuration hardware. |
| Hardware_Ver | Char(16) | Version of the configuration hardware. |
| Loaded_conf | Num(10) | Loaded configuration. |
| Conf_Load_Time | Num(10) | Time the configuration was loaded. |
| Conf_Loader | Char(64) | Loader used to load the configuration. |
| Max_Slots | Num(2) | Maximum number of slots in a router. |
| Available_Slots | Num(2) | Number of slots which are empty. |

**SCANNER**

The Scanner table contains information about a scanner which is attached to a PC.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a scanner. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a scanner. Relates to *UID* in the Device Information table. |
| Gray_Scale | Num(3) | Number of gray scales. |
| Colors | Num(8) | Number of colors. |
| Min_Resolution | Num(4) | Min resolution in dpi. |
| Max_Resolution | Num(4) | Max resolution in dpi. |
| Type | Char(9) | Type of scanner (flat-bed or hand-held). |
| Port_UID* | Num(10) | Identifies the port on a device which is associated with a Scanner. Relates to *UID* in the Port table. |

**SEGMENT**

The Segment table contains information about the organization's network segments. A segment is the lowest level network view. One or more segments are connected to form a subnet.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of the segment. |
| Segment_Name | Char(25) | Segment's common name. |
| Topography | Char(10) | Topography of a segment (e.g. Bus). |
| Medium_Access_Control | Char(10) | Medium access control being used (Token ring). |
| Physical_Aspect | Char(30) | Physical aspect of a segment (e.g. broadband coaxial). |
| Administrator_UID* | Num(10) | Identifies a segment's administrator. Relates to *UID* in the Administrator table. |
| Subnet_UID* | Num(10) | Identifies a subnet which is associated with a segment. Relates to *UID* in the Subnet table. |

**SLOT**

The Slot table contains information about slots associated with a device.

| Field Name | Type | Description |
|---|---|---|
| Slot_Number | Num(2) | Unique identifier of a slot. |
| Bus_Type | Char(10) | Type of Bus used. |
| Bus_Transfer_Rate | Num(5) | Transfer rate in KB/sec. |
| Card_Description | Char(25) | Description of the card that is in the slot (e.g. Ethernet). |

## SOFTWARE INFORMATION

The Software Information table contains information which is common to all software applications.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a software application's common information. |
| Name | Char(15) | Name of a software application. |
| Version | Char(5) | Version number of a software application. |
| Developer | Char(15) | Name of the developer of a software application. |
| Site_License_Number | Num(10) | Number of the site license. |
| Allowed_Copies | Num(3) | Number of allowed copies of a software application per the site license agreement. |
| Renewal_Date | Date | Date the site license needs to be renewed. |
| Tech_Assist_Num | Char(12) | Phone number to call and obtain technical assistance with a software application. |
| Purchase_Reqn_Num* | Char(16) | Identifies the purchase order which a software application was procured. Relates to *Requisition_Number* in the Purchase table. |

## SPACE

The Space table contains information about a space where network components are located.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a space. |
| Name | Char(25) | Name of the space. |
| Room | Num(4) | Room number of the space. |
| Access_Requirements | Char(25) | Access requirements of the space (i.e. key badge, combination lock, none, etc.). |

| | | |
|---|---|---|
| Building_UID* | Num(10) | Identifies the building in which a space is located. Relates to *UID* in the Building table. |
| Notes | Memo | Any additional information about a space that may be useful. |

**STAND ALONE PC**

The Stand Alone PC table contains information about a stand alone PC.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a PC. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a stand alone PC. Relates to *UID* in the Device Information table. |
| PC_Name | Char(25) | Name of the PC (e.g. Zenith 486 Slimline). |
| Bus_Type | Char(4) | Bus type (e.g. ISA). |
| Processor | Char(3) | Processor (e.g. 386). |
| Processor_Speed | Num(3) | Processor speed in MHz. |
| Co_Processor_Flag | Logical | Identifies if there is a co-processor attached (.T.). |
| Keyboard_Type | Char(10) | Type of keyboard (e.g. 101-key IBM). |
| Gameport_Flag | Logical | Identifies if there is a game port attached (.T.). |
| Mouse | Char(15) | Type of mouse attached (e.g. Logitech). |
| BIOS | Char(10) | Type of BIOS installed. |
| BIOS_Date | Date | The date of the BIOS. |
| Base_Memory | Num(3) | Amount of conventional memory installed in KB. |
| Extended_Memory | Num(5) | Amount of extended memory installed in KB. |
| Expanded_Memory | Num(5) | Amount of expanded memory installed in KB. |
| Alternate_Function | Char(15) | Alternate function the PC is performing (e.g. Print Server) |

## STAR COUPLER

The Star Coupler table contains information about a star coupler which connects network nodes.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a star coupler. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a star coupler. Relates to *UID* in the Device Information table. |
| Segment_UID* | Num(10) | Identifies the segment to which a star coupler is a part of. Relates to *UID* in the Segment table. |

## SUBNET

The Subnet table contains information about the organization's subnets. One or more subnets are connected to form a network. A subnet is made up of one or more segments.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a subnet. |
| Name | Char(25) | Subnet's common name. |
| Topography | Char(10) | Topography of the subnet (i.e. bus, star, ring, etc.). |
| Administrator_UID* | Num(10) | Identifies a subnet's administrator. Relates to *UID* in the Administrator table. |
| Network_UID* | Num(10) | Identifies the network associated with a subnet. Relates to *UID* in the Network table. |

## SUN TERMINAL

The SUN Terminal table contains information about a SUN Terminal.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a SUN Terminal. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a SUN terminal. Relates to *UID* in the Device Information table. |
| Keyboard_Type | Char(15) | Type of keyboard (e.g. 101-key IBM). |

## SUPERNET

The Supernet table contains information about the organization's networks as a whole. This is a top level view of the organization's network system.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of the organization's network system. |
| Name | Char(25) | Organization's name. |
| Topography | Char(10) | Overall topography of the organization's networks (e.g. FDDI Ring). |
| Administrator_UID* | Num(10) | Identifies a supernet's administrator. Relates to *UID* in the Administrator table. |

## TAPE DRIVE

The Tape Drive table contains information about a tape drive.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a tape drive. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a tape |

| | | |
|---|---|---|
| | | drive. Relates to *UID* in the Device Information table. |
| Capacity | Num(4) | Capacity of a tape drive in MB. |
| Transfer_Rate | Num(4) | Transfer rate in KB/sec. |
| Tape_Server_UID* | Num(10) | Identifies a tape server which is associated with a tape drive. Relates to *UID* in the Tape Server table. |
| Port_UID* | Num(10) | Identifies the port on a device which is associated with a Tape Drive. Relates to *UID* in the Port table. |

**TAPE SERVER**

The Tape Server table contains information about a tape server which provides user access to tape drives attached to the network and runs file backup and archiving programs that can service the entire network.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a tape server. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a tape server. Relates to *UID* in the Device Information table. |

**TERMINAL BLOCK**

The Terminal Block table contains information about a terminal block.

| Field Name | Type | Description |
|---|---|---|
| Block_Code | Char(1) | Unique identifier of a terminal block. |
| Cable_Code | Char(3) | Code used to identify the cable in a terminal block. |
| Direct_Line | Logical | Identifies if a terminal block is a direct line (.T.). |
| DDN | Logical | Identifies if a terminal block is a Defense Data Network (DDN) line. |

## TERMINAL SERVER

The Terminal Server table contains information about a terminal server which connects a large number of terminals to IBM host systems or minicomputer systems over a local area network (LAN).

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a terminal server. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a terminal server. Relates to *UID* in the Device Information table. |

## TRANCEIVER

The Tranceiver table contains information about tranceivers which provide the interface between a node and the network.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a tranceiver. |
| Type | Char(10) | Type of tranceiver. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a tranceiver. Relates to *UID* in the Device Information table. |

## USER

The User table contains information about each user of a circuit.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a user. |
| Name | Char(25) | Name of the individual user (e.g. NAVOCEANO). |

## USER GROUP

The User Group table contains information about the various groups of users that use a circuit.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a user group. |
| Name | Char(10) | Name of the group (e.g. Red Tess) |

## USER GROUP / USER RELATION

The User Group / User Relation table provides the link for a many-to-many relationship between the User Group table and the User table.

| Field Name | Type | Description |
|---|---|---|
| User_Group_UID* | Num(10) | Identifies a group of users. Relates to UID in the User Group table. |
| User_UID* | Num(10) | Identifies a user of a circuit. Relates to UID in the User table. |

## VOLUME

The Volume table contains information about the volume on a hard disk which is associated with a file server.

| Field Name | Type | Description |
|---|---|---|
| Volume_Number | Num(10) | Unique identifier of a volume. |
| Volume_Name | Char(10) | Name of a disk volume. |
| Size | Num(4) | Size of a volume in MB. |
| Starting_Block | Num(10) | Starting block of a volume. |
| Total_Blocks | Num(10) | Total number of blocks in a volume. |

## WARRANTY

The Warranty table contains warranty information about network components.

| Field Name | Type | Description |
|---|---|---|
| Warranty_Number | Num(16) | Unique identifier of a maintenance contract. |
| Start_Date | Date | Date the warranty goes in to effect. |
| End_Date | Date | Date the warranty expires. |
| Type_of_Warranty | Char(25) | Type of warranty service which is provided on network components. |
| Company_UID* | Num(10) | Identifies the company which provides warranty services. Relates to *UID* in the Company table. |

## XTERMINAL

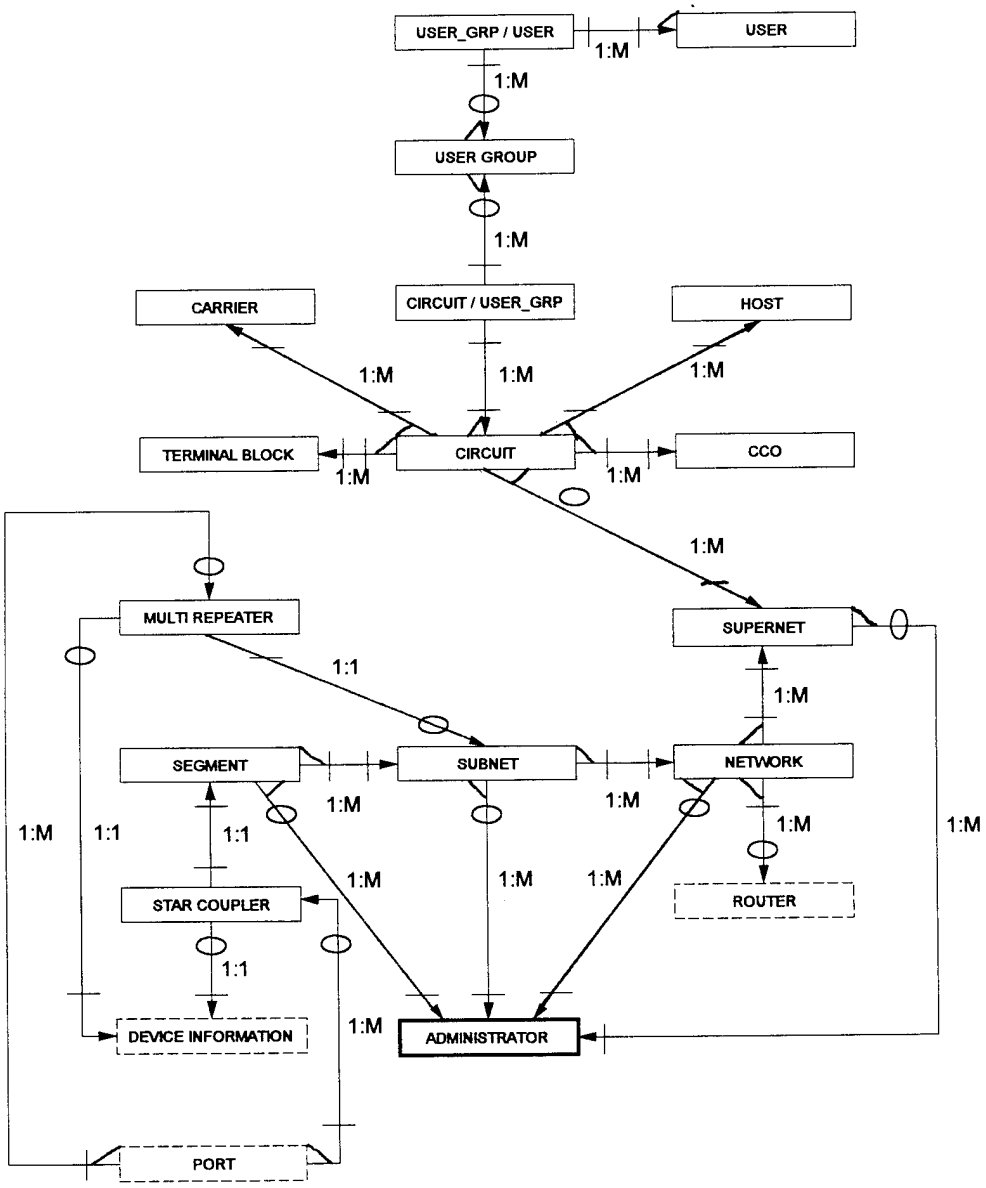
The XTerminal table contains information about a XTerminal.

| Field Name | Type | Description |
|---|---|---|
| UID | Num(10) | Unique identifier of a XTerminal. |
| Device_Info_UID* | Num(10) | Identifies the common device information associated with a XTerminal. Relates to *UID* in the Device Information table. |
| Keyboard_Type | Char(10) | Type of keyboard (e.g. 101-key IBM). |

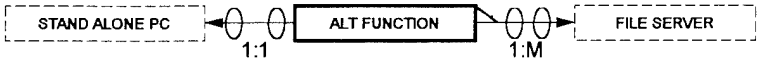
# APPENDIX B. ENTITY-RELATIONSHIPS

An Entity-Relationship (ER) diagram describes the overall structure of the data relationships which are being modeled. Each entity, denoted by a rectangle, represents a separate data table. Rectangles with bold lines represent the central entity of the diagram. Those with dashed lines represent the central entity of another diagram by the same name.

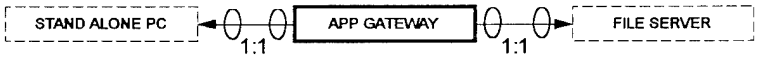
## ADMINISTRATOR DIAGRAM



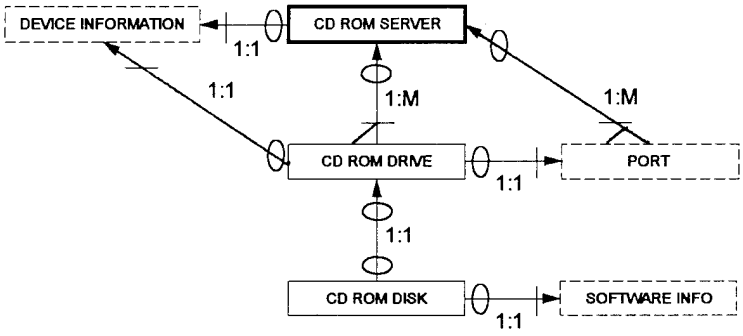
## ALTERNATE FUNCTION DIAGRAM




## APPLICATION GATEWAY DIAGRAM

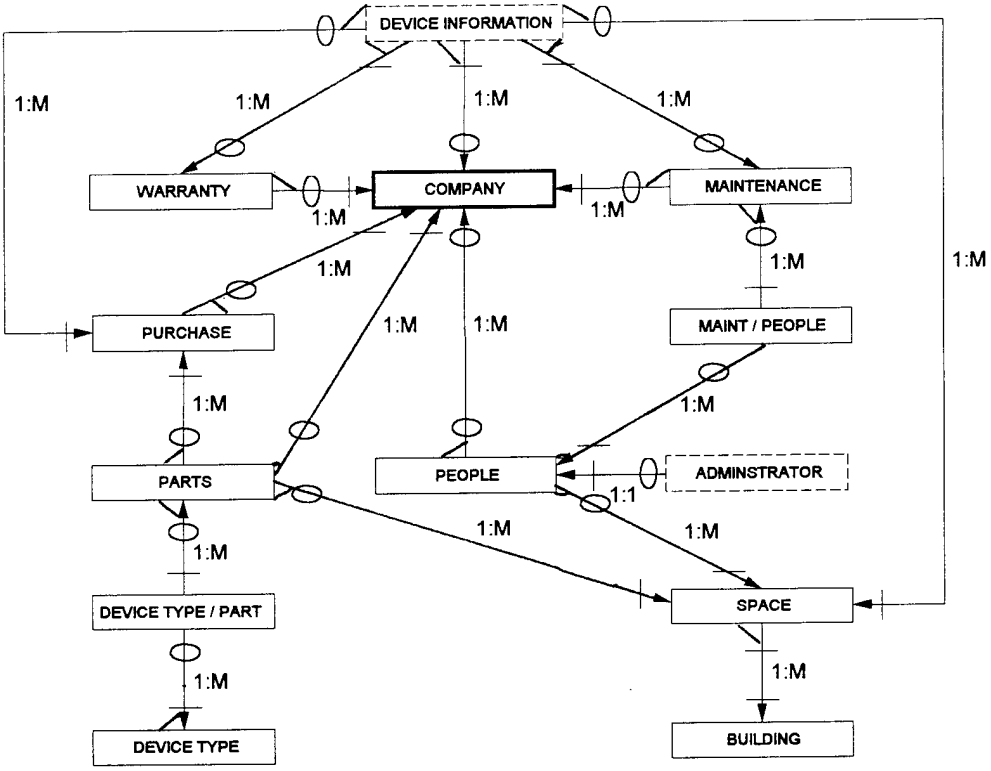



## CD ROM SERVER DIAGRAM

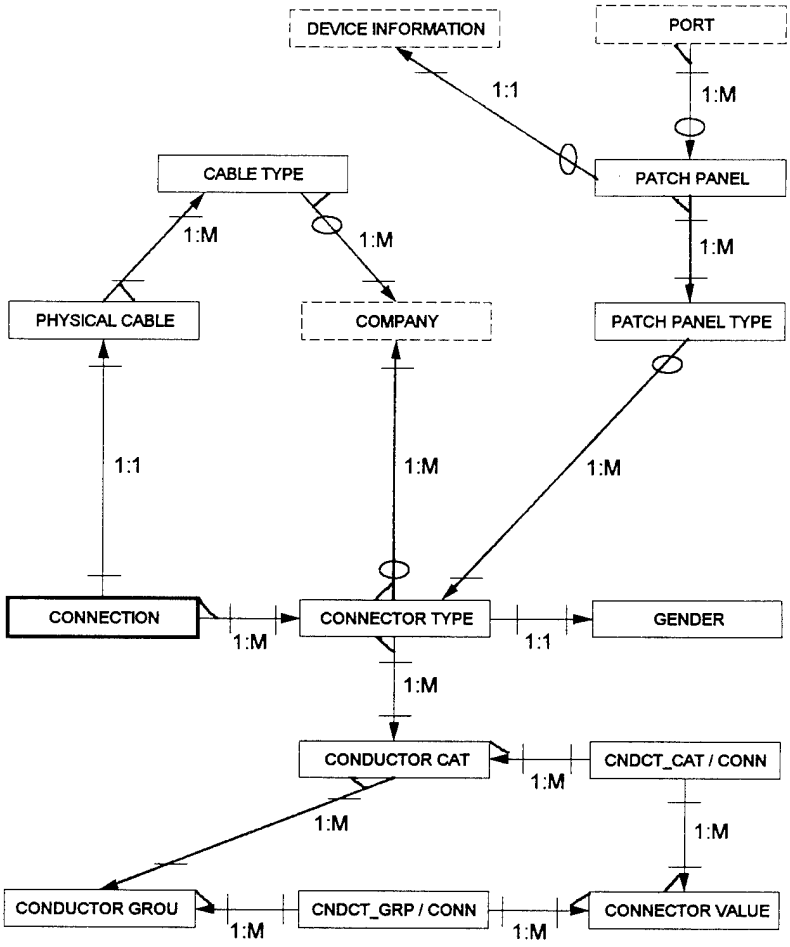



## COMMUNICATION SERVER DIAGRAM

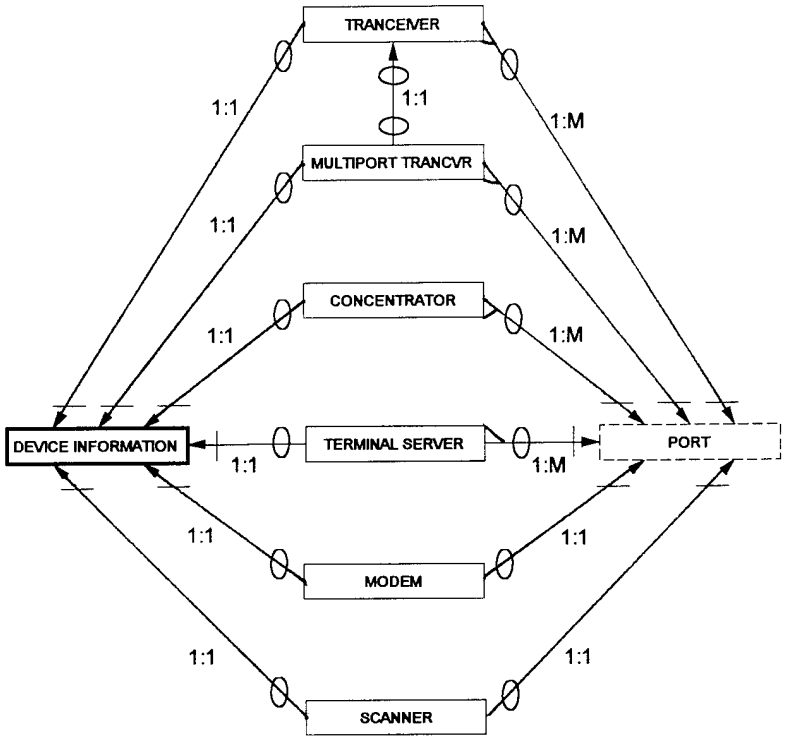
## COMPANY DIAGRAM



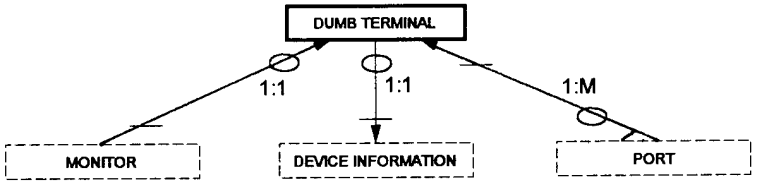
## CONNECTION DIAGRAM



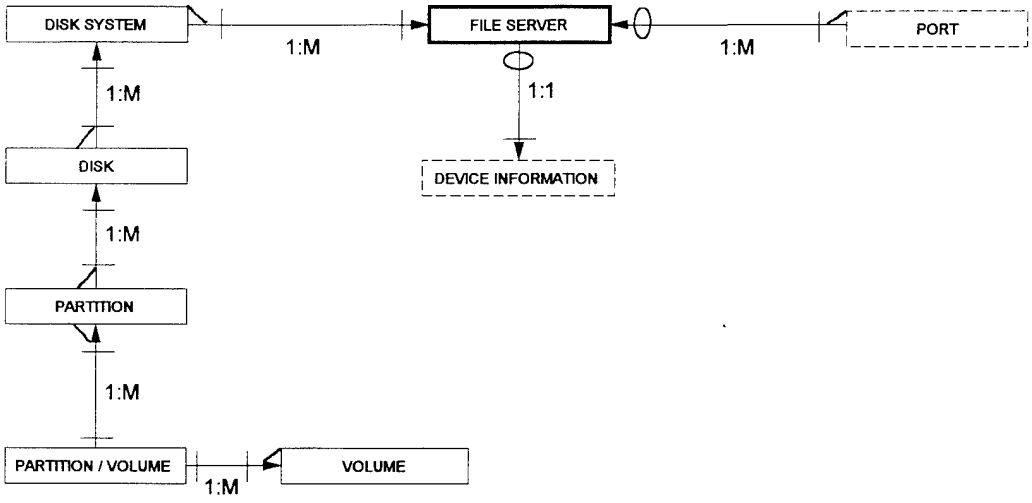
## DEVICE INFORMATION DIAGRAM



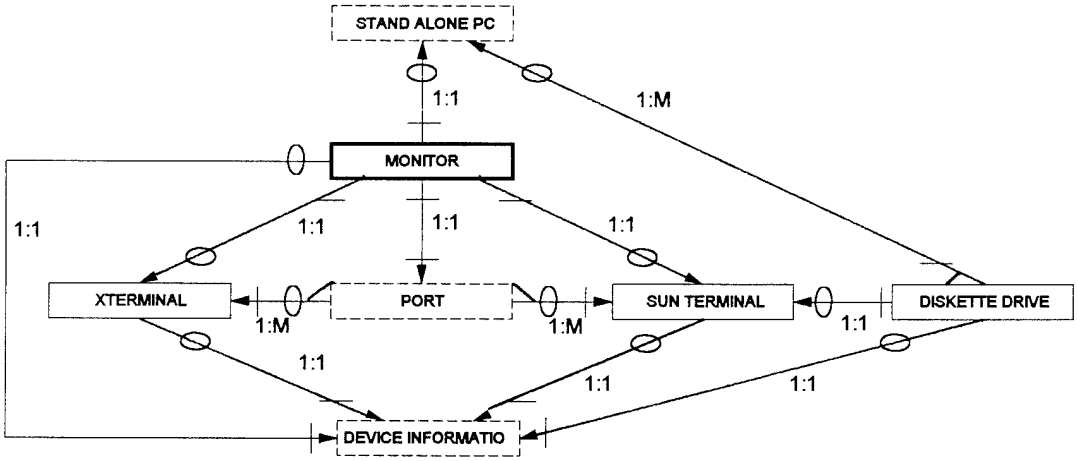

## DUMB TERMINAL DIAGRAM



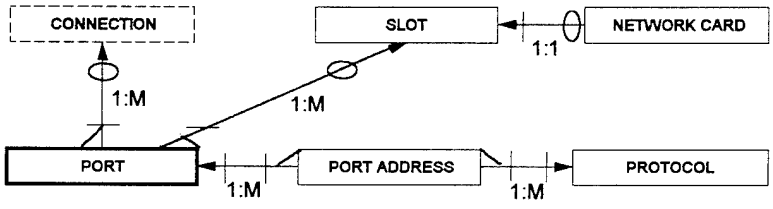
## FILE SERVER DIAGRAM



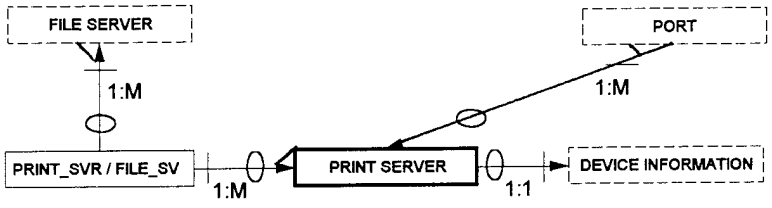

## MONITOR DIAGRAM



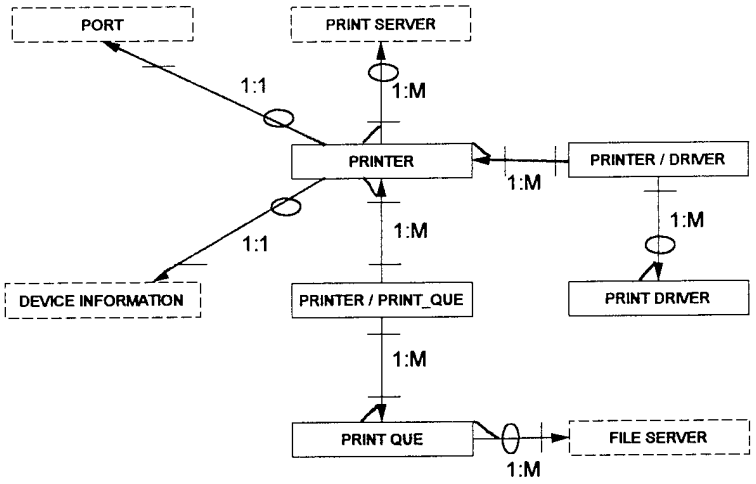

## PORT DIAGRAM



## PRINT SERVER DIAGRAM

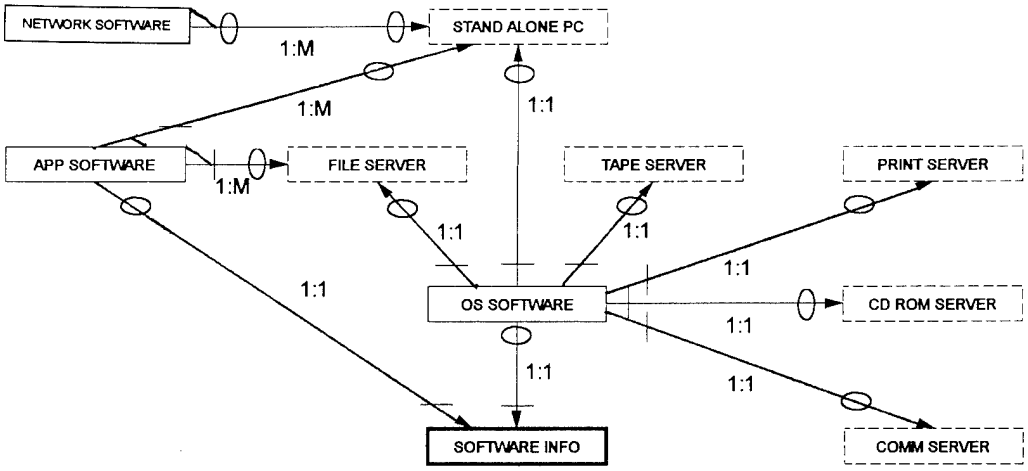



## PRINTER DIAGRAM

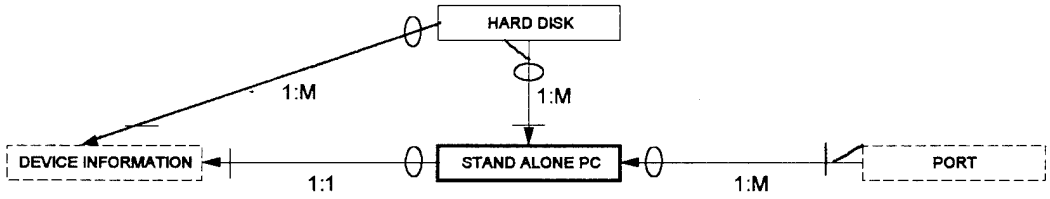



## ROUTER DIAGRAM

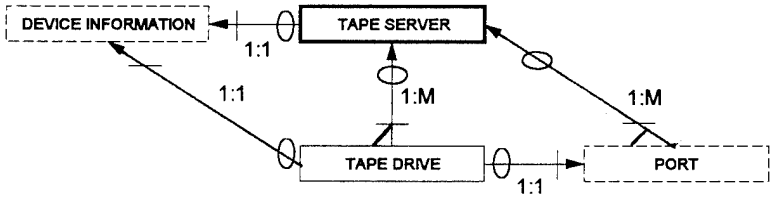
## SOFTWARE INFORMATION DIAGRAM




## STAND ALONE PC DIAGRAM




## TAPE SERVER DIAGRAM



# LIST OF REFERENCES


1. Duff, Jame B., Hunter, James D. and Matthews, David C., "Process Management: The Future of Integrated Management Systems," *Integrated Network Management, III*, Elsevier Science Publishers B.V., 1993

2. Cornetti, Richard, "Special Report: the case for proactive network management," *The LocalNetter*, vol. 14 no. 3, March 1994

3. Liebmann, Lenny, "Network Management: Getting a Grip," *LAN Magazine*, vol. 9, no. 13, December 1994

4. Kapoor, Atul, "SNMP Based Network Management - A Report on the State of the Art," Kaptronix Inc., June 1993

5. Stallings, William, *SNMP, SNMPv2, and CMIP - The Practical Guide to Network-Management Standards*, Addison & Wesley, 1993

6. Haritsa, Jayant, Ball, Michael, Roussopoulos, Nicholas, Baras, John, and Datta, Anindya, "Design of the MANDATE MIB," *Integrated Network Management, III*, Elsevier Science Publishers B.V., 1993

7. Disabato, Michael C., "Key Technologies for Integrated Network Management," *Integrated Network Management, III*, Elsevier Science Publishers B.V., 1993

8. DKSystems Inc., *DKInventory Manager for Windows™ - User Guide*, 1993

9. Auto-trol Technology Corporation, *KONFIG™ User's Reference Release 2.2*, 1994

10. Hong, James W., Bauer, Michael A., and Bennett, J. Michael, "Integration of the Directory Service in the Network Management Framework," *Integrated Network Management, III*, Elsevier Science Publishers B.V., 1993

11. Neumair, B., "Modelling Resources for Integrated Performance Management," *Integrated Network Management, III*, Elsevier Science Publishers B.V., 1993

12. Meandzija, Branislav and Zimmer, Wolfgang, "Strategies for the Nineties," *Integrated Network Management, III*, Elsevier Science Publishers B.V., 1993

# INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center ..... 2
   Cameron Station
   Alexandria, Virginia 22304-6145

2. Library, Code 52 ..... 2
   Naval Postgraduate School
   Monterey, California 93943- 5101

3. Professor Myung Suh Code SM/SU ..... 1
   Department of Systems Management
   Naval Postgraduate School
   Monterey, California 93943-5000

4. Commander William B. Short Code SM/SH ..... 1
   Department of Systems Management
   Naval Postgraduate School
   Monterey, California 93943-5000

5. FLENUMMETOCCEN ..... 5
   7 Grace Hopper Avenue
   Attn: Code 53
   Monterey, California 93943-5501